

Received: xxx Accepted: xxx Published: xxx.

DOI: [10.30473/coam.2026.78581.1439](https://doi.org/10.30473/coam.2026.78581.1439)

Volume xxx, Issue xxx, p.p. 1-??: xxx

Research Article



Open Access

Control and Optimization in Applied Mathematics - COAM

Demand-Aware Heuristic Initialization and NSGA-II for Multi-Objective Transit Route Optimization

Reza Rajabi¹, Farzin Modarres Khiabani¹ ✉, Saba Sadeghi Gavgani²

¹ Department of Mathematics, Ta.c., Islamic. Azad University, Tabriz, Iran

² Department of Mathematics, Sara.C., Islamic. Azad University, Sarab, Iran

✉ Correspondence:

Farzin Modarres Khiabani

E-mail:

f.modarres@iaui.ac.ir

How to Cite

Rajabi, R., Modarres Khiabani, F., Sadeghi Gavgani, S. (2027). "Demand-aware heuristic initialization and NSGA-II for multi-objective transit route optimization". *Control and Optimization in Applied Mathematics*, X(x), 1-??.
<https://doi.org/10.30473/coam.2026.78581.1439>

Abstract. Candidate route sets in the Urban Transit Network Design Problem (UTNDP) must provide adequate network coverage and connectivity while complying with prescribed route-length and route-cardinality requirements. This study develops a demand-aware heuristic-initialized NSGA-II framework for the UTNDP. The method uses demand and travel-time information to construct feasible initial route sets, followed by Pareto ranking, crowding-distance selection, crossover, problem-specific mutation, feasibility repair, and duplicate filtering within the evolutionary search. The optimization model minimizes two explicitly defined objectives: the demand-weighted passenger generalized travel time (F_1) and the total route weight, which serves as an operator-cost proxy (F_2). In the reported Mandl/Mumford-style benchmark evaluation, every transfer edge of the transfer-expanded graph carries a fixed 5-min penalty, consistent with the standard convention for the urban transit routing problem (UTRP). Candidate route sets that leave any positive-demand origin–destination pair disconnected are first subjected to feasibility repair and, if the repair fails, are rejected before environmental selection. The framework is assessed on established UTNDP benchmark networks ranging from small to large instances. Hypervolume and spacing are formally defined as Pareto-set performance indicators, whereas the reported numerical comparisons are restricted to benchmark objective values directly supported by the available computational records; no run-level inferential comparison is claimed. Duplicate filtering and parameter settings are treated as algorithmic design components rather than being assigned independent gains without a dedicated ablation study. On the reported benchmark data, the framework yields passenger-cost values of 21.34 min for Mumford0 (22.18 min in the original study by Mumford) and 52.18 min for Edinburgh200 (55.34 min in the original study by Mumford). Frequency and vehicle capacity are treated as exogenous quantities so that the experiments isolate route-set optimization. The resulting framework provides a reproducible basis for passenger–operator trade-off analysis and a clearly delimited platform for future extensions to joint frequency, capacity, uncertainty, and parallel-computing formulations.

Keywords. Urban transit network design, Multi-objective optimization, NSGA-II, Pareto optimization, Heuristic initialization, Hypervolume.

MSC. 90C29; 90C27; 90B06.

<https://mathco.journals.pnu.ac.ir>

©2026 by the authors. Licensee PNU, Tehran, Iran. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 International (CC BY4.0) (<http://creativecommons.org/licenses/by/4.0>)

1 Introduction

In the Urban Transit Network Design Problem (UTNDP), a given set of stops must be organized into routes that serve passenger demand efficiently without imposing excessive route cost on the operator. Even when stop locations are fixed, identifying an appropriate route configuration remains computationally difficult: the search space expands rapidly with network size, and every admissible solution must satisfy coverage, connectivity, route-length, and route-cardinality requirements. The present study addresses this route-design problem by embedding a demand-informed initialization procedure within a Non-dominated Sorting Genetic Algorithm II (NSGA-II) search process.

Transit service planning is commonly decomposed into network design, frequency setting, timetable development, vehicle scheduling, and crew scheduling [?]. These stages are interdependent, but their joint optimization substantially enlarges the decision space. The present study therefore isolates the route-design stage: stop locations, origin–destination (OD) demand, service frequency, and vehicle capacity are treated as exogenous inputs, while the route set is optimized under the structural constraints defined in Section ??.

The UTNDP is characterized by conflicting objectives: passengers prefer short travel times, few transfers, short waiting times, and extensive coverage, whereas operators are concerned with minimizing operating costs and ensuring efficient resource utilization [?, ?]. Balancing these perspectives necessitates a multi-objective formulation, in which solutions are evaluated according to Pareto efficiency rather than a single global optimum [?, ?]. As a result, the problem is widely recognized as combinatorial, nonlinear, and NP-hard [?, ?], and the size of the feasible search space grows exponentially with the number of candidate stops, routes, and demand interactions. While exact algorithms such as branch-and-bound and mixed-integer programming can yield optimal solutions for small-scale networks, they quickly become infeasible for large real-world cases [?, ?].

To address these computational challenges, researchers have turned extensively to meta-heuristic and evolutionary optimization algorithms capable of efficiently exploring large, discontinuous search spaces. Among early applications, Baker and Ayeche [?] demonstrated the capability of the genetic algorithm (GA) to reduce routing costs in vehicle routing problems. Baldacci et al. [?] proposed a set-partitioning approach for exact optimization, and Bell and McMullen [?] investigated ant colony optimization (ACO) for routing, which exhibits strong convergence properties. Brands and van Berkum [?] extended evolutionary techniques to multi-modal network design, confirming their effectiveness in balancing divergent objectives such as cost and service quality. Related routing research also established complementary heuristic and evolutionary foundations. Clarke and Wright [?] introduced the savings heuristic for vehicle scheduling from a central depot, and Gillett and Miller [?] introduced the sweep heuristic for constrained vehicle-dispatch problems, while Tan et al. [?, ?] developed hybrid multi-objective

evolutionary approaches for vehicle-routing problems with time-window and truck-and-trailer constraints. Parallel metaheuristic search was also investigated by Czech and Czarnas [?] through a simulated-annealing approach to the vehicle-routing problem with time windows.

Numerous subsequent studies explored variations of these algorithms under realistic urban settings. Almasi et al. [?] optimized the Daejeon transit network in South Korea under variable demand using single- and multi-objective heuristics. Jha et al. [?] and Momenitabar and Mattson [?] jointly optimized route structure and frequency, highlighting the importance of integrating operator and passenger interests. Cooper et al. [?] and John et al. [?] exploited parallel multi-objective evolutionary algorithms for large-scale UTNDPs, achieving considerable computational improvements.

Among metaheuristics, NSGA-II has become particularly prominent owing to its balance between exploration and elitism. Chai and Liang [?] enhanced NSGA-II for the simultaneous optimization of routes and frequencies, reporting significant reductions in both travel time and operating cost. Chen et al. [?] adapted it to a low-carbon, multi-period location-routing framework; Carvajal-Carreño et al. [?] and Zhang et al. [?] used NSGA-II variants for energy-efficient metro operation; and Miandoabchi et al. [?] as well as Seifpour et al. [?] extended hybrid NSGA-II models to multimodal and stochastic contexts. Evolutionary multi-objective optimization has also been applied to multimodal passenger routing: Farooqi [?] used NSGA-II for route finding in a multimode transportation network, while Qi et al. [?] addressed integrated shared-taxi and bus route planning through a nondominated-sorting multi-objective genetic framework. Beyond transportation routing, Li et al. [?] applied NSGA-II within a parametric multi-objective framework for urban-block morphology optimization, further illustrating the adaptability of the algorithm across complex urban design problems. Collectively, these studies demonstrate the versatility of NSGA-II in addressing complex multi-objective design problems in which decision interdependencies are substantial.

Recent peer-reviewed work has materially advanced the same problem family. Hüsselmann et al. [?] combine improved initialization, new perturbation operators, hyperheuristic operator management, and an NSGA-II variant for the bi-objective UTRP. Zervas et al. [?] develop a parallel artificial-bee-colony/hill-climbing method for large-scale UTRP instances, while Kourepinis et al. [?] propose a discrete artificial-fish-swarm formulation with explicit computational-cost reduction. Holliday et al. [?] use deep reinforcement learning and graph neural networks to learn route-improvement heuristics on Mandl/Mumford-type benchmarks, and Yoo et al. [?] jointly optimize route design and frequency using reinforcement learning. Martinez-Quezada et al. [?] extend NSGA-II to an electric transit network design and frequency-setting formulation with charging decisions. Viewed against these developments, the contribution of the present study lies in how several problem-specific mechanisms are organized within a single route-design procedure. Demand-aware weighted-graph construction is used to generate initial solutions, feasibility is enforced through repair or rejection, route changes are introduced

through a defined operator set, and exact edge-set duplicates are controlled before standard NSGA-II environmental selection. The study does not modify the underlying non-dominated sorting mechanism; rather, its methodological contribution concerns the integration and explicit specification of these UTNDP-specific components.

The passenger side of transit planning cannot, however, be reduced entirely to network-level optimization criteria. Studies of travel behavior have shown that service reliability, comfort, and individual attitudes also shape the willingness to use public transport, as reported by Stradling et al. [?], Gardner and Abraham [?], and Beirão and Cabral [?]. These findings emphasize that quantifiable service metrics, such as travel time or cost, cannot alone explain users' mode choices; perceived quality and user experience are equally decisive. Thus, passenger-centric criteria should be embedded directly within the optimization framework rather than considered ex post.

Despite substantial progress, reproducibility and methodological traceability remain important concerns in UTNDP research. Models differ in objective definitions, feasibility rules, benchmark conventions, and treatment of operational variables, which makes direct numerical comparison difficult. In particular, route-design studies should distinguish the objectives actually optimized from related service attributes such as transfer count, frequency, and capacity, unless those variables are explicitly represented in the mathematical model. The present formulation therefore focuses on two consistent objectives, namely passenger travel time and operator route cost, under structural route-set constraints.

In this context, metaheuristics remain a compelling solution framework. Their stochastic yet guided search mechanisms make them capable of identifying near-optimal solutions in high-dimensional spaces where traditional methods fail [?, ?]. Widely used approaches, such as genetic algorithms [?], tabu search [?], ant colony optimization [?], and hybrid metaheuristics [?, ?], have each contributed to expanding the frontier of feasible UTNDP applications.

Building upon this foundation, the present study considers the redesign of a fixed-stop urban transit network as a bi-objective UTNDP. Throughout the paper, the computational method is termed the demand-aware heuristic-initialized NSGA-II framework. It minimizes demand-weighted passenger travel time and operator route-weight cost while enforcing route coverage, route-length bounds, transfer connectivity, global connectivity, and a prescribed number of routes. Frequency setting, vehicle capacity, and fleet scheduling remain outside the decision space of the present model.

The contribution of this study is centered on the formulation and implementation of a traceable route-design procedure rather than on a modification of the core NSGA-II algorithm. At the modeling level, passenger generalized travel time and operator route cost are treated as separate objectives and are evaluated only after route-set feasibility has been established. At the solution-construction level, demand-aware initialization is coupled with coverage and connectivity repair, allowing the evolutionary search to operate over route sets that satisfy the pre-

scribed structural requirements. The NSGA-II search is further equipped with route-specific variation operators and deterministic edge-set duplicate control, while inherited algorithmic elements remain explicitly distinguished from the components introduced in the present implementation.

The framework is examined on established UTNDP benchmark networks using a common definition of the multi-objective performance indicators and an instance-specific reference-point convention. This experimental structure permits comparisons across network sizes while retaining a clear correspondence between the mathematical formulation, the algorithmic components, and the reported results. From a planning perspective, the method is intended as a route-design decision-support framework for examining passenger-operator trade-offs in existing urban transit networks; broader operational objectives such as frequency, capacity, and fleet decisions remain outside the present optimization model.

The remainder of this paper is organized as follows. Section ?? presents the problem definition and mathematical formulation. Section ?? documents the Bee Colony Optimization baseline that serves as methodological context. Section ?? develops the demand-aware heuristic-initialized NSGA-II framework and reports the experimental design and results. Section ?? concludes the paper and outlines directions for future research.

2 Problem Definition and Formulation

The mathematical model is developed below from the network representation and route-feasibility conditions to the passenger and operator objective functions. The same section also defines the evaluation quantities and benchmark conventions required for the subsequent computational analysis.

2.1 Multi-Objective Optimization Background

Let S denote the feasible decision set and let $f: S \rightarrow \mathbb{R}^k$ be the objective vector, $f(x) = (f_1(x), f_2(x), \dots, f_k(x))$. The generic multi-objective minimization problem is written as

$$\min_{x \in S} \{f_1(x), f_2(x), \dots, f_k(x)\}.$$

A feasible solution x' dominates a feasible solution x if the condition in Equation (??) holds:

$$f_i(x') \leq f_i(x) \quad \forall i = 1, \dots, k, \quad \text{and} \quad \exists j : f_j(x') < f_j(x). \quad (1)$$

A feasible solution is Pareto optimal if no other feasible solution dominates it; that is, no other feasible solution improves at least one objective without worsening another. All non-dominated

feasible solutions form the Pareto set, whose image in the objective space defines the Pareto front. For the UTNDP considered here, this trade-off is represented by two objectives associated with passenger and operator costs.

2.2 Network Representation

The underlying urban public transport infrastructure is modeled as a connected, weighted graph

$$G = (V, E, W), \quad (2)$$

where

- $V = \{v_1, v_2, \dots, v_n\}$ is the set of stops (nodes), with $|V| = n$;
- $E = \{e_1, e_2, \dots, e_m\} \subseteq V \times V$ is the set of road segments (edges), with $|E| = m$;
- $W = \{\omega_1, \omega_2, \dots, \omega_m\}$, with $\omega_j > 0$, is the set of edge travel-time weights, where ω_j is the time required to traverse edge e_j .

Passenger demand is represented by an origin–destination (OD) matrix

$$D = [D_{v_i, v_j}]_{n \times n}, \quad (3)$$

where D_{v_i, v_j} denotes the passenger demand from origin v_i to destination v_j during the planning period. The benchmark travel-time and OD-demand matrices are treated as fixed inputs. The notation used throughout the paper is summarized in Table ??.

Remark 1. Throughout the paper, R_i denotes an individual route, $\mathcal{R}$ a route set, r the prescribed route count, and n_T the node count of the transfer-expanded transit graph. The hypervolume definition in Section ?? uses a separate symbol for the reference point.

2.3 Route Representation

Within G , each route R_i is represented as a simple path: consecutive stops are connected by edges in E , and a stop may not appear more than once on the same route. We use V_{R_i} and E_{R_i} for the stop and edge sets associated with R_i , respectively. A complete candidate solution consists of the collection of routes

$$\mathcal{R} = \{R_1, R_2, \dots, R_r\}. \quad (4)$$

Here, r is the prescribed number of routes to be operated. For a given benchmark design, r is fixed during an optimization run, and every accepted route set must satisfy the cardinality constraint in Equation (??).

Table 1: Notation summary.

Symbol	Definition
$G = (V, E, W)$	Weighted graph representing the transit network
V	Set of nodes (stops), $ V = n$
E	Set of edges (road segments), $ E = m$
ω_j	Weight (travel time) of edge e_j
D_{v_i, v_j}	Passenger demand from node v_i to node v_j
R_i	A single route (simple path in G)
$\mathcal{R} = \{R_1, \dots, R_r\}$	Route set (candidate solution)
V_{R_i}, E_{R_i}	Nodes and edges belonging to route R_i
m_1, m_2	Minimum and maximum route-length bounds
$\alpha_{v_i, v_j}(\mathcal{R})$	Minimum travel time from v_i to v_j under route set $\mathcal{R}$
$F_1(\mathcal{R})$	Passenger cost objective (average travel time)
$F_2(\mathcal{R})$	Operator route-weight objective (sum of route edge weights; operating-effort proxy)
$G_{\mathcal{R}}$	Subgraph induced by route set $\mathcal{R}$
n_T	Number of nodes in the transfer-expanded transit graph

2.4 Feasibility Constraints

A route set $\mathcal{R}$ must satisfy the following structural constraints to be operationally feasible.

Constraint 1 (network coverage). Every stop must be served by at least one route:

$$\bigcup_{i=1}^{|\mathcal{R}|} V_{R_i} = V. \quad (5)$$

Constraint 2 (route-length bounds). The number of stops on each route must lie within pre-defined limits:

$$m_1 \leq |V_{R_i}| \leq m_2, \quad \forall R_i \in \mathcal{R}, \quad (6)$$

where m_1 and m_2 are the prescribed minimum and maximum numbers of stops on a route. These bounds control route extent and operational manageability; vehicle-capacity effects are not represented by these bounds in the present model.

Constraint 3 (transfer connectivity). Each route must share at least one stop with another route to enable passenger transfers:

$$\forall R_i \in \mathcal{R}, \exists R_j \in \mathcal{R}, j \neq i: V_{R_i} \cap V_{R_j} \neq \emptyset. \quad (7)$$

Equation (??) is a local route-adjacency requirement only: by itself, it does not guarantee that all routes belong to one transfer-connected component. System-wide connectivity is enforced by Equation (??), and passenger-level feasibility is additionally verified by requiring a finite generalized path for every positive-demand OD pair in $\mathcal{P}^+$ (defined in Section ??) before objective evaluation.

Constraint 4 (global connectivity). The subgraph induced by all routes must be connected:

$$G_{\mathcal{R}} = \left(\bigcup_{i=1}^{|\mathcal{R}|} V_{R_i}, \bigcup_{i=1}^{|\mathcal{R}|} E_{R_i} \right) \text{ is connected.} \quad (8)$$

Constraint 5 (route-set cardinality). The number of routes is fixed for a given design:

$$|\mathcal{R}| = r. \quad (9)$$

In practice, additional realistic constraints (e.g., fleet-size bounds, a maximum number of transfers, or coverage of high-demand nodes) may be enforced as hard constraints or incorporated through penalty terms in the objectives.

2.5 Objective Functions

Passengers are assigned to minimum-generalized-time paths in the transfer-expanded network. Let $\alpha_{v_i, v_j}(\mathcal{R})$ denote the minimum finite generalized travel time between origin v_i and destination v_j under route set $\mathcal{R}$. In the reported benchmark implementation, an in-route edge retains its original travel-time weight, whereas each transfer edge is assigned a fixed penalty $\tau_{\text{tr}} = 5$ min [?, ?]. Thus, $\alpha_{v_i, v_j}(\mathcal{R})$ includes the in-vehicle travel time plus 5 min for every route transfer along the selected path. The passenger objective is the demand-weighted average generalized travel time:

$$F_1(\mathcal{R}) = \frac{\sum_{i=1}^n \sum_{j=1}^n D_{v_i, v_j} \alpha_{v_i, v_j}(\mathcal{R})}{\sum_{i=1}^n \sum_{j=1}^n D_{v_i, v_j}}. \quad (10)$$

Equation (??) therefore penalizes both long in-vehicle travel and excessive transfers through the explicit 5-min transfer-edge weight. It is evaluated only for feasible route sets. Let $\mathcal{P}^+$ denote the set of OD pairs with strictly positive demand. A route set is passenger-feasible only if $\alpha_{v_i, v_j}(\mathcal{R})$ is finite for every $(v_i, v_j) \in \mathcal{P}^+$. Offspring violating this condition are sent to the coverage/connectivity repair procedure; if feasibility is not restored, they are rejected before NSGA-II environmental selection.

The second objective aggregates the edge weights of all operated routes. Because the benchmark edge weights represent travel times, F_2 is interpreted as a route-weight proxy for operating effort rather than as a monetary cost. Under the fixed-frequency assumption, it is defined as

$$F_2(\mathcal{R}) = \sum_{i=1}^{|\mathcal{R}|} \sum_{e_j \in E_{R_i}} \omega_j. \quad (11)$$

Accordingly, Equation (??) is used strictly as a time-weighted route-load proxy for operator effort under fixed service frequency; it is not interpreted as a monetary operating cost or a physical distance.

Combining the two objectives, the UTNDP is formulated as the bi-objective minimization problem

$$\min_{\mathcal{R}} (F_1(\mathcal{R}), F_2(\mathcal{R}))$$

subject to constraints (??)–(??). Due to the conflicting nature of the objectives (longer routes may reduce passenger travel time but increase operator cost), the goal is to approximate the Pareto front of non-dominated solutions rather than a single optimum.

2.6 Transit Network Evaluation

To evaluate the passenger objective, a transfer-expanded transit graph is constructed from each route set $\mathcal{R}$. A physical stop that belongs to more than one route is represented by one route-specific node per occurrence. Consecutive stops on the same route are connected with the original travel-time weights, while route-specific copies of the same physical stop are connected by transfer edges of weight $\tau_{tr} = 5$ min. Standard shortest-path computation on this graph therefore accounts explicitly for transfer inconvenience rather than treating transfers as zero-cost links.

The all-pairs shortest path (APSP) problem on the transit network is solved to obtain travel times between all pairs of original stops. The Floyd–Warshall algorithm is employed for this purpose.

2.6.1 Computational Challenge

Route-set evaluation requires repeated shortest-path calculations on a transfer-expanded transit graph. Because the same evaluation is performed for many candidate route sets and generations, the computational cost increases rapidly with benchmark size. Table ?? therefore reports the base-node sizes of the benchmark suite used to assess scalability; no unverified transit-node growth factors are assumed.

The benchmark sequence spans one order of magnitude in base network size, from Mandl (15 nodes) to Edinburgh200 (200 nodes). This scale difference, together with repeated feasibility checks and all-pairs path evaluation, motivates the use of population-based metaheuristics and a fixed computational budget for comparative experiments.

Table 2: Benchmark instances and network sizes.

Instance	Nodes	Benchmark role	Primary source
Mandl	15	Canonical small network	Mandl [?]
Mumford0	30	Synthetic benchmark	Mumford [?]
Mumford1	70	Synthetic benchmark	Mumford [?]
Mumford2	110	Synthetic benchmark	Mumford [?]
Mumford3	127	Synthetic benchmark	Mumford [?]
Nottingham100	100	Large-scale benchmark	Cooper et al. [?]
Edinburgh200	200	Large-scale benchmark	Cooper et al. [?]

2.7 Benchmark Instances

Seven established UTNDP benchmark instances are considered: Mandl, Mumford0–Mumford3, Nottingham100, and Edinburgh200. The Mandl network represents the canonical small test instance, the Mumford family provides progressively larger synthetic benchmarks, and Nottingham100/Edinburgh200 provide large-scale stress tests used in prior evolutionary UTNDP studies [?, ?, ?]. Their node counts are reported in Table ??.

The benchmark data are used in their established published form; the present study does not claim to have regenerated the Mandl or Mumford networks from random spatial sampling. Travel-time and OD-demand matrices are treated as fixed benchmark inputs for each instance, which enables a reproducible comparison of route-set optimization methods.

For the larger Nottingham100 and Edinburgh200 instances, the analysis likewise uses the published benchmark representations rather than reconstructing stop locations or OD demand from external map or population services. This distinction avoids conflating benchmark evaluation with a separate empirical demand-estimation study.

2.8 Performance Metrics

Two standard multi-objective performance indicators are defined for characterizing approximated Pareto fronts: spacing, for distributional diversity, and hypervolume, for joint convergence and diversity. In the present study, these indicators establish the methodological evaluation framework; however, complete run-level hypervolume and spacing distributions are not available for all configurations and are therefore not used to support inferential statistical claims.

2.8.1 Spacing Metric (Diversity)

The spacing metric Δ measures the uniformity of the solution distribution along the Pareto front. For an obtained non-dominated set P with solutions $x_1, \dots, x_{|P|}$,

$$\Delta = \sqrt{\frac{1}{|P| - 1} \sum_{i=1}^{|P|} (\bar{d} - d_i)^2},$$

$$d_i = \min_{j \neq i} (|F_1(x_i) - F_1(x_j)| + |F_2(x_i) - F_2(x_j)|). \quad (12)$$

Here, d_i is the nearest-neighbor distance in the bi-objective space defined by Equation (??), and $\bar{d}$ is the arithmetic mean of these distances. A spacing value of zero indicates perfectly uniform nearest-neighbor spacing.

2.8.2 Hypervolume Metric (Convergence and Diversity)

The hypervolume indicator in Equation (??) measures the objective-space volume dominated by a non-dominated set P relative to one instance-specific reference point z^{ref} . The reference point is chosen componentwise worse than every solution included in that instance-level comparison, and its exact construction is specified in Section ??.

$$S(P; z^{\text{ref}}) = \text{vol} \left(\bigcup_{u \in P} E(u, z^{\text{ref}}) \right), \quad (13)$$

where

$$E(u, z^{\text{ref}}) = \{v \in \mathbb{R}^2 \mid u_1 \leq v_1 \leq z_1^{\text{ref}}, \\ u_2 \leq v_2 \leq z_2^{\text{ref}}\}.$$

Larger hypervolume values indicate a better approximation of the true Pareto front, simultaneously capturing both convergence and diversity.

3 Bee Colony Optimization Baseline and Methodological Context

This section documents the Bee Colony Optimization (BCO) route-modification mechanism solely as a baseline and methodological reference. The mathematical expressions and route-modification mechanisms presented in Sections ??–??, including the two bee types, the normalized solution-quality score, and the exponential loyalty rule, follow the transit-network BCO formulation of Nikolić and Teodorović [?]. They are not claimed as new components of the proposed NSGA-II framework; the present contribution begins with the demand-aware initialization and evolutionary integration in Section ??.

Remark 2. Methodological attribution to Nikolić and Teodorović [?]: the Type 1 terminal replacement, the Type 2 terminal extension/deletion, the direct-service score, and the loyalty mechanism presented in Sections ??–?? are reproduced or notation-adapted from the published BCO baseline. Their role here is comparative: they establish a transparent prior-work reference against which the construction and NSGA-II framework of Section ?? are distinguished.

3.1 Initial Solution Generation

Consider a baseline bus line l with terminal stops i and j . The line contains the stops on the shortest path connecting i and j ; let N_l denote the set of stops served by that line. This bus line can be used both by passengers traveling directly and by those who must make transfers during their journey. The total number of passengers ds_{ij} who benefit from direct service on this bus line, that is, the direct-service score associated with the terminal pair, is

$$ds_{ij} = \sum_{m \in N_l} \sum_{n \in N_l} D_{mn}. \quad (14)$$

In Equation (??), D_{mn} is the OD demand from stop m to stop n , and both summations are taken over the stops served by line l . We define the corresponding matrix DS containing the number of passengers benefiting from direct service:

$$DS = \{ds_{ij}\}, \quad i, j \in \{1, 2, \dots, |V|\}.$$

A simple procedure for generating an initial solution is presented in Algorithm ?. It attempts to connect node pairs with high ds_{ij} values using direct services, thereby increasing the number of passengers who benefit from direct travel.

3.2 Solution Improvement

In the BCO baseline, solution improvement is performed through forward passes using two classes of bees with distinct route modifications, as in Nikolić and Teodorović [?]. Type 1 bees modify a selected line by replacing one terminal and reconstructing the line through a shortest path; Type 2 bees shorten or extend a selected line at a terminal. Thus, the Type 1/Type 2 distinction is reproduced as prior methodology. The present study does not claim priority for heterogeneous bees; its methodological focus is the demand-aware initialization, the NSGA-II environmental selection, the feasibility repair, and the integrated route-operator framework described in Section ??.

- Type 1 bees: prior-work terminal-replacement modification [?].

Algorithm 1 BCO baseline initial-solution generation [?]

Input: Network G , direct-service matrix DS , target number of bus lines N_{BL} , and feasibility constraints (??)–(??).

Output: A feasible baseline route set $\mathcal{V}$, or an infeasibility flag if the repair is unsuccessful.

- 1: $\mathcal{V} \leftarrow \emptyset$; set $q \leftarrow 1$.
- 2: **while** $q \leq N_{BL}$ **do**
- 3: Select the terminal pair (a, b) with the largest current direct-service score ds_{ab} .
- 4: Construct the shortest-path route R_{ab} and update $\mathcal{V} \leftarrow \mathcal{V} \cup \{R_{ab}\}$.
- 5: Update DS by removing the passenger demand already served directly by the current route set.
- 6: $q \leftarrow q + 1$.
- 7: **end while**
- 8: Evaluate $\mathcal{V}$ against constraints (??)–(??) and the positive-demand OD-connectivity rule.
- 9: **if** any feasibility condition is violated **then**
- 10: Apply the same coverage/connectivity repair logic used for candidate route sets; if the repair fails, return an infeasibility flag.
- 11: **end if**
- 12: **return** $\mathcal{V}$.

- Type 2 bees: prior-work terminal extension/deletion modification [?].

The two bee types differ only in their route-improvement procedure; this distinction is inherited from the published BCO formulation [?].

3.2.1 Solution Improvement by Type 1 Bees

Consider a given set L of bus lines. A Type 1 bee selects one line from L . The probability of selecting line l is

$$P_l = \frac{ds_{ij}}{\sum_{q \in L} ds_{a_q b_q}},$$

where

- i and j are the terminal stops of the selected line l ;
- L is the current set of bus lines;
- for each $q \in L$, the symbols a_q and b_q denote the two terminal stops of line q ;
- ds_{ij} is the total direct-service demand associated with the selected terminal pair.

Suppose a Type 1 bee selects line l with terminal stops i and j . One terminal is retained; assume i is retained. The other terminal is replaced by a candidate stop k sampled according to Equation (??), and the modified line is reconstructed as a shortest path from i to k :

$$P_k = \frac{ds_{ik}}{\sum_{t \in V} ds_{it}}. \quad (15)$$

Once the new terminal k is selected, the shortest path between i and k defines the replacement bus line.

3.2.2 Solution Improvement by Type 2 Bees

A Type 2 bee selects a line using the same line-selection probability as the Type 1 bee and then chooses one of its two terminals uniformly at random for local modification. The selected terminal is either removed, shortening the line by one stop, or retained while the line is extended to one adjacent feasible stop, according to the baseline BCO modification rule [?].

3.3 Comparison Mechanism for Partial Solutions

After improving their solutions, both Type 1 and Type 2 bees return to the hive. All improved solutions generated by all bees are then evaluated. Following Nikolić and Teodorović [?], let T_b denote the total passenger travel time of the network generated by bee b . Equation (??) reproduces the baseline normalization that maps this value to a quality score over the current bee population:

$$O_b = \frac{T_{\max} - T_b}{T_{\max} - T_{\min}}, \quad b = 1, 2, \dots, B. \quad (16)$$

Here, $T_{\max}$ and $T_{\min}$ are, respectively, the largest and smallest total passenger travel times among the candidate networks generated in the current bee population.

Equation (??) is likewise the exponential loyalty rule of the documented BCO baseline [?]. Let $O_{\max}$ denote the largest normalized quality score in the current bee population. The probability that bee b remains loyal to its previously generated solution is

$$P_b = \exp[-(O_{\max} - O_b)], \quad b = 1, 2, \dots, B. \quad (17)$$

Using Equation (??) and a uniform random draw, each bee either retains its current solution or becomes an uncommitted follower. The retention probability increases monotonically with normalized quality; consequently, the best normalized solution has loyalty probability one, and lower-quality solutions have smaller retention probabilities, consistent with the BCO baseline [?].

3.4 Scope, Planning Context, and Methodological Boundary

The optimization implemented in this paper is a route-design problem. Service frequency and vehicle capacity are not decision variables in Equations (??)–(??), and no frequency-optimization or capacity-constrained Pareto result is claimed. A fixed-frequency, sufficient-capacity interpretation is used when F_2 is related to operator effort. Integrated route–frequency–capacity optimization is therefore reserved for future work rather than presented as part of the current algorithm.

Commercial transport-planning systems provide network representation, demand assignment, simulation, and decision-support functions, but this paper does not evaluate software products. Their detailed feature-by-feature discussion is therefore omitted because it is not part of the mathematical contribution; the present focus is automated route-set optimization.

This section has documented the BCO route-modification baseline from the prior literature [?] and clarified that frequency setting, capacity optimization, and commercial planning software are outside the computational contribution of this paper. The proposed computational study begins in Section ?? with a demand-aware heuristic-initialized NSGA-II route-design framework.

4 Demand-Aware Heuristic-Initialized NSGA-II for Transit Network Design

4.1 Introduction

The UTNDP is NP-hard, with conflicting objectives and a rapidly expanding feasible route-set space. Genetic and evolutionary methods are well suited to such structural search because they can modify complete route sets while preserving multiple non-dominated alternatives; early transit-network GA studies include Chakroborty and Dwivedi [?], while later benchmark developments and operators are documented by Mumford [?] and John et al. [?].

However, generating feasible solutions for the UTNDP remains challenging because multiple constraints must be satisfied simultaneously (Equations (??)–(??) in Section ??). While heuristic methods have been extensively employed for initial solution generation, many existing approaches generate an excessive number of routes, leading to prohibitive computational costs for larger instances.

The computational framework evaluated in this section integrates the following components; it is presented as a transparent combination of established multi-objective selection and problem-specific route-design components rather than as a new named metaheuristic family:

- a demand-aware heuristic construction method for generating feasible initial populations;

- the established NSGA-II framework for Pareto ranking, elitism, and crowding-distance selection [?];
- eight route-level mutation operators, combining established and adapted UTNDP modifications [?, ?, ?];
- an edge-based diversity measure and deterministic duplicate filtering used as population-management safeguards.

The objective is to approximate the trade-off between passenger cost (demand-weighted travel time) and operator route cost as defined in Equations (??)–(??), subject to the feasibility conditions in Section ?. Figure ?? summarizes the resulting computational workflow.

Within each generation, selected parents undergo crossover and route-level mutation. Infeasible offspring are subjected to feasibility repair and are rejected if the repair fails. Feasible offspring are then evaluated with respect to F_1 and F_2 , exact duplicate route sets are filtered using the edge-set criterion, and parent–offspring environmental selection is performed using non-dominated rank and crowding distance. This process is repeated until $G_{\max} = 200$ generations are completed, as summarized in Figure ??.

4.2 Heuristic Initial Population Generation

4.2.1 Weighted Graph Combination

Let G be the transit graph defined in Equation (??), with travel-time edge weights W , and let D be the OD-demand matrix defined in Equation (??). For each edge $e = (u, v)$, the travel-time term and the associated OD-demand term are normalized to $[0, 1]$ by Equation (??):

$$\begin{aligned}\omega'_j &= \frac{\omega_j - \omega_{\min}}{\omega_{\max} - \omega_{\min}}, \\ D'_{uv} &= \frac{D_{uv} - D_{\min}}{D_{\max} - D_{\min}}.\end{aligned}\tag{18}$$

The normalization removes scale differences before construction scoring. Equation (??) uses the inverse normalized demand $1 - D'_{uv}$ because route construction repeatedly selects low-weight feasible edges: a high-demand edge must therefore receive a smaller demand contribution if it is to be preferentially attracted into a seed route set. The score is a construction weight only and is not an objective scalarization of F_1 and F_2 :

$$w_j^{(i)} = \lambda_1 \omega'_j + \lambda_2 (1 - D'_{uv}), \quad \forall e_j = (u, v) \in E,\tag{19}$$

where i indexes the coefficient pair (λ_1, λ_2) .

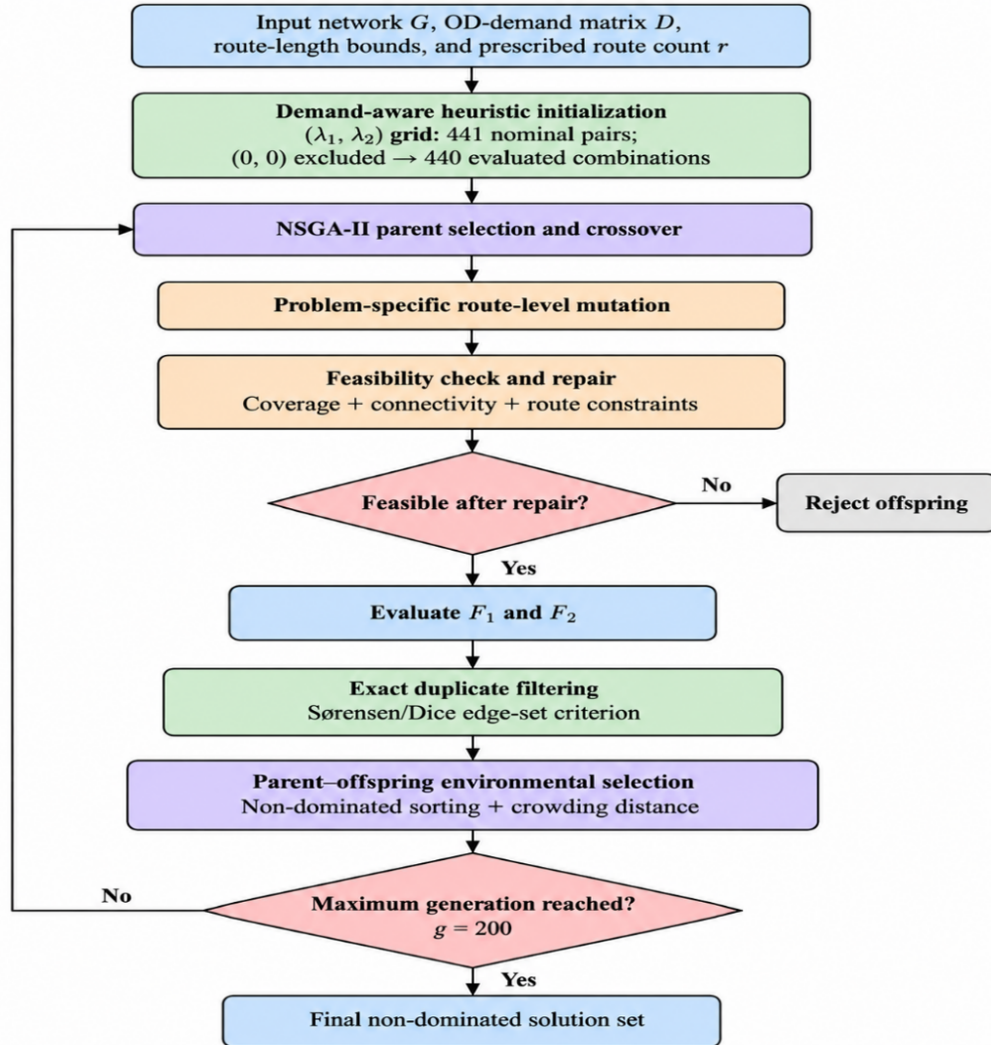


Figure 1: Computational workflow of the demand-aware heuristic-initialized NSGA-II framework.

The coefficients λ_1 and λ_2 each take values in $\{0, 0.05, 0.10, \dots, 1\}$, yielding 441 nominal grid pairs. The degenerate pair $(0, 0)$, which assigns zero weight to every edge, is skipped; the remaining 440 pairs are evaluated to generate candidate seed route sets. Infeasible and exact-duplicate candidates are discarded. If more than N feasible unique seeds are obtained, non-dominated rank and crowding distance on (F_1, F_2) are used to retain N well-spread candidates; if fewer than N remain, demand-guided route generation and admissible route perturbation are used to complete the population with feasible unique seeds. The coefficient grid is therefore screened exhaustively rather than sampled randomly.

4.2.2 Spanning Subgraph Construction

For each coefficient pair (λ_1, λ_2) , a spanning route subgraph is constructed while enforcing the connectivity and route-cardinality requirements in Equations (??)–(??). Algorithm ?? begins from a low-weight feasible edge and extends routes with minimum-weight feasible edges subject to the route-length bounds in Equation (??). If an exact r -route spanning construction is not obtained from the current coefficient pair, the repair/route-generation mechanism (Algorithm ??) is invoked rather than accepting an infeasible route set.

Algorithm 2 Heuristic construction of the route set $\mathcal{R}$ for a weighted graph G_i

Input: Weighted graph $G_i = (V, E, W_i)$, route-length bounds m_1, m_2 , target route count r .

```

1:  $\mathcal{R} \leftarrow \emptyset$ 
2: while  $\bigcup_{R \in \mathcal{R}} V_R \neq V$  do
3:   if  $|\mathcal{R}| = 0$  then
4:     Select the lowest-cost edge  $e = (u, v) \in E_i$  and create the route  $R \leftarrow \{u, v\}$ .
5:   else
6:     Select the lowest-cost edge sharing a vertex with the existing routes and create the
       route  $R$  from it.
7:   end if
8:   while  $|V_R| < m_2$  do
9:     Extend  $R$  by adding the lowest-cost feasible edge.
10:  end while
11:   $\mathcal{R} \leftarrow \mathcal{R} \cup \{R\}$ 
12: end while
13: if  $|\mathcal{R}| < r$  then
14:   Apply demand-guided route generation (Algorithm ??) to add the remaining routes.
15: end if
16: return  $\mathcal{R}$ 

```

Algorithm 3 Demand-guided route generation, adapted from established UTNDP construction procedures [?, ?]

Input: Current route set $\mathcal{R}$, graph G , demand matrix D .

- 1: Identify the vertex pairs (v_i, v_j) not directly served by $\mathcal{R}$.
- 2: Sort the pairs in descending order of demand D_{v_i, v_j} .
- 3: **for** each pair (v_i, v_j) **do**
- 4: Compute the k -shortest paths between v_i and v_j .
- 5: **for** each candidate route R **do**
- 6: **if** R is feasible and $\sum_{e \in E_R} \omega_e < \alpha_{v_i, v_j}(\mathcal{R})$ **then**
- 7: **return** R
- 8: **end if**
- 9: **end for**
- 10: **end for**
- 11: **return** $\emptyset$

4.3 NSGA-II Framework for UTNDP

NSGA-II [?] is used as the environmental-selection framework for the bi-objective UTNDP, as outlined in Algorithm ???. The implementation follows non-dominated sorting, elitist parent-offspring replacement, crowding-distance preservation, and binary tournament selection. SEAMO2-based configurations from prior UTNDP work [?] are retained only as methodological comparators where their definitions are explicitly traceable.

4.4 Genetic Operators

4.4.1 Crossover

We adopt Mumford's crossover operator [?], which constructs offspring by selecting routes from parents so as to maximize the coverage of unseen vertices. For each selected route, the proportion of unseen vertices, $|V_{\text{unseen}}|/|V_{R_i}|$, is computed, and routes maximizing this ratio are selected until $|\mathcal{R}'| = r$. If the coverage constraint (??) is violated, a repair procedure attempts to add the missing vertices to route terminals by extending routes while maintaining simplicity and the length bounds.

Algorithm 4 NSGA-II for the UTNDP**Input:** Population size N , maximum number of generations $G_{\max}$.

- 1: $P_0 \leftarrow$ heuristic initial population (Section ??)
- 2: Evaluate the objectives F_1, F_2 for all $p \in P_0$.
- 3: $t \leftarrow 0$
- 4: **while** $t < G_{\max}$ **do**
- 5: $Q_t \leftarrow$ offspring population generated via selection, crossover, and mutation
- 6: $U_t \leftarrow P_t \cup Q_t$
- 7: $\{\mathcal{F}_1, \mathcal{F}_2, \dots\} \leftarrow \text{fast-non-dominated-sort}(U_t)$
- 8: $P_{t+1} \leftarrow \emptyset, i \leftarrow 1$
- 9: **while** $|P_{t+1}| + |\mathcal{F}_i| \leq N$ **do**
- 10: crowding-distance-assignment($\mathcal{F}_i$)
- 11: $P_{t+1} \leftarrow P_{t+1} \cup \mathcal{F}_i$
- 12: $i \leftarrow i + 1$
- 13: **end while**
- 14: Sort $\mathcal{F}_i$ by crowding distance.
- 15: $P_{t+1} \leftarrow P_{t+1} \cup \mathcal{F}_i[1 : (N - |P_{t+1}|)]$
- 16: $t \leftarrow t + 1$
- 17: **end while**
- 18: **return** the non-dominated solutions of $P_{G_{\max}}$

4.4.2 Mutation Operators

The number of mutations per offspring follows a binomial distribution with r trials and success probability $1/r$. Eight mutation operators are employed:

1. **Add Nodes:** randomly adds up to $I \leq m_2/2$ vertices to route terminals while maintaining feasibility.
2. **Del Nodes:** randomly removes up to $I \leq m_2/2$ vertices from route terminals.
3. **Exchange** [?, ?]: selects a route R_i and a crossing route at a shared vertex and swaps segments to create two new routes.
4. **Merge** (Algorithm ??): merges two routes sharing a terminal vertex, provided the merged route remains simple and satisfies the length bounds.
5. **Replace** (Algorithm ??): removes the route serving the lowest demand and replaces it using Algorithm ??.
6. **Remove-Overlapping** (Algorithm ??): removes routes that are subsequences of other routes.
7. **2-opt** [?]: randomly selects two vertices within a route and reverses the sequence between them.
8. **Invert-Exchange** (Algorithm ??): selects two routes and two valid index positions, inverts the vertex sequences between the indices, and exchanges them between the routes.

Remark 3. Fallback rule: if the Merge operator finds no pair of routes satisfying the shared-terminal, simplicity, and route-length conditions, the mutation is treated as a no-op, and the original feasible route set is returned unchanged. The search therefore continues without generating an infeasible offspring or forcing a structurally invalid merge.

4.5 Population Diversity Measure

For a route set $\mathcal{R}$, we define the edge multiset $E_{\mathcal{R}} = \{\{u, v\} : (u, v) \in E_{R_i}, R_i \in \mathcal{R}\}$, which captures all edges in $\mathcal{R}$. The Sørensen/Dice similarity coefficient between two route sets $\mathcal{R}^{(a)}$ and $\mathcal{R}^{(b)}$ is

$$\text{Sim}(\mathcal{R}^{(a)}, \mathcal{R}^{(b)}) = \frac{2 |E_{\mathcal{R}^{(a)}} \cap E_{\mathcal{R}^{(b)}}|}{|E_{\mathcal{R}^{(a)}}| + |E_{\mathcal{R}^{(b)}}|}.$$

Population diversity is then computed as

Algorithm 5 Merge operator for the route set $\mathcal{R}$

```

1: mutated  $\leftarrow$  false
2: repeat
3:   Randomly select  $R \in \mathcal{R}$  without replacement.
4:   for each  $R_i \in \mathcal{R}$  do
5:     if  $R$  and  $R_i$  share a terminal and  $|R \cap R_i| = 1$  then
6:       Remove  $R$  and  $R_i$  from  $\mathcal{R}$ .
7:        $R_{\text{merged}} \leftarrow$  merge of  $R$  and  $R_i$  at the common terminal.
8:        $\mathcal{R} \leftarrow \mathcal{R} \cup \{R_{\text{merged}}\}$ 
9:       Apply route generation (Algorithm ??) to add a replacement route.
10:      mutated  $\leftarrow$  true
11:      break
12:    end if
13:  end for
14: until no routes remain or mutated = true
15: if  $\mathcal{R}$  is feasible then
16:   return  $\mathcal{R}$ 
17: else
18:   return the original solution
19: end if

```

Algorithm 6 Replace operator for the route set $\mathcal{R}$

```

1: Compute the direct passenger demand served by each route.
2: Remove the route with the lowest demand from  $\mathcal{R}$ .
3: Generate a replacement route using Algorithm ?? and insert it into  $\mathcal{R}$ .
4: if  $\mathcal{R}$  is infeasible then
5:   Repair  $\mathcal{R}$ .
6: end if
7: return  $\mathcal{R}$ 

```

Algorithm 7 Remove-overlapping operator for the route set $\mathcal{R}$

```

1:  $mutated \leftarrow \text{false}$ 
2: repeat
3:   Randomly select  $R \in \mathcal{R}$  without replacement.
4:   for each  $R_i \in \mathcal{R}$  do
5:     if  $R$  is a subsequence of  $R_i$  then
6:       Remove  $R$  from  $\mathcal{R}$ .
7:        $mutated \leftarrow \text{true}$ 
8:       break
9:     end if
10:  end for
11: until no routes remain or  $mutated = \text{true}$ 
12: if  $\mathcal{R}$  is feasible then
13:   return  $\mathcal{R}$ 
14: else
15:   return the original solution
16: end if

```

Algorithm 8 Invert-exchange operator for the route set $\mathcal{R}$

```

1:  $mutated \leftarrow \text{false}$ 
2: repeat
3:   Randomly select  $R_1, R_2 \in \mathcal{R}$  without replacement.
4:   Generate random indices  $start, end$  with  $start < end \leq \min(|R_1|, |R_2|)$ .
5:    $R_1^{inv} \leftarrow$  inverted segment of  $R_1$  between  $start$  and  $end$ .
6:    $R_2^{inv} \leftarrow$  inverted segment of  $R_2$  between  $start$  and  $end$ .
7:    $R_1[start : end] \leftarrow R_2^{inv}$ 
8:    $R_2[start : end] \leftarrow R_1^{inv}$ 
9:   if  $\mathcal{R}$  is feasible then
10:     $mutated \leftarrow \text{true}$ 
11:   else
12:    Revert the changes.
13:   end if
14: until no routes remain or  $mutated = \text{true}$ 
15: return  $\mathcal{R}$ 

```

$$\text{Diversity}(P) = \frac{2}{|P|(|P| - 1)} \sum_{i=1}^{|P|-1} \sum_{j=i+1}^{|P|} (1 - \text{Sim}(\mathcal{R}^{(i)}, \mathcal{R}^{(j)})),$$

where $\mathcal{R}^{(i)}$ denotes the i th route set of the population P . This measure monitors population similarity during evolution. Exact duplicate route sets (zero pairwise edge-set distance) are removed as a deterministic population-hygiene step inside the NSGA-II implementation. Duplicate filtering is not treated as an independent fifth metaheuristic variant, and no separate performance gain is attributed to it without an explicit ablation experiment.

Computational complexity. Let n_T denote the number of nodes in the transfer-expanded transit graph, N the population size, $G_{\max}$ the number of generations, and $M = 2$ the number of objectives. The Floyd–Warshall algorithm requires $\mathcal{O}(n_T^3)$ time and $\mathcal{O}(n_T^2)$ memory for one route-set evaluation and is therefore the principal computational bottleneck as route overlap enlarges the transfer-expanded graph. Across the evolutionary search, shortest-path evaluation contributes $\mathcal{O}(G_{\max} N n_T^3)$, compared with $\mathcal{O}(G_{\max} M N^2)$ for non-dominated sorting and $\mathcal{O}(G_{\max} M N \log N)$ for crowding-distance ordering. The initializer additionally evaluates a finite grid of weighted construction graphs. This complexity analysis motivates future investigation of sparse repeated-Dijkstra/Johnson-type APSP schemes, cached substructure, and topology-aware shortest-path acceleration.

4.6 Experimental Design

Four experimentally identifiable algorithm configurations are distinguished:

- Configuration A: SEAMO2 framework with Mumford-style operators [?] and random initialization;
- Configuration B: SEAMO2 framework with demand-aware heuristic initialization;
- Configuration C: SEAMO2 framework with demand-aware initialization and the combined route-operator set;
- Configuration D: NSGA-II [?] with demand-aware initialization, the combined route-operator set, feasibility repair, and deterministic duplicate filtering.

These configurations and their component traceability are summarized in Table ??.

The comparative protocol uses a population size of $N = 200$, a maximum of $G_{\max} = 200$ generations, and a crossover probability of $p_c = 0.9$. For methodological consistency, the hypervolume reference point is defined componentwise as 1.2 times the largest objective value included in the corresponding instance-level comparison. The same reference-point convention is applied whenever Pareto sets are compared within an instance.

Table 3: Experimental algorithm configurations and component traceability.

Configuration	Selection framework	Initialization	Route operators	Duplicate handling
A	SEAMO2	Random	Mumford-style [?]	No separate filter
B	SEAMO2	Demand-aware heuristic	Baseline operator set	No separate filter
C	SEAMO2	Demand-aware heuristic	Combined route operators	No separate filter
D	NSGA-II [?]	Demand-aware heuristic	Combined route operators	Deterministic filter

Parameter rationale. The population size $N = 200$, the generation limit $G_{\max} = 200$, and the crossover probability $p_c = 0.9$ are fixed a priori and applied identically to all algorithm configurations so that component comparisons use the same computational budget. The mutation probability is route-count adaptive, $p_m = 1/r$, so the expected number of route-level mutation events remains approximately one per offspring. No post-hoc automated tuner (e.g., SMAC or iRace) is used to optimize these values against the reported benchmark outcomes; consequently, the study evaluates the framework under a controlled common parameter regime rather than claiming globally optimal hyperparameters.

Statistical reporting. The numerical benchmark results reported in the present paper are treated descriptively. Because a complete matched archive of run-level hypervolume and spacing outcomes is not available for all four configurations, no Wilcoxon, Friedman, or other inferential significance test is reported. Accordingly, the numerical results reported in Section ?? are interpreted descriptively, and no claim of statistical superiority is made.

4.7 Results and Discussion

4.7.1 Benchmark Passenger-Cost Comparison

Table ?? reports selected best-found passenger-cost objective values from the current computational study alongside literature values stated under comparable benchmark names. These values are used solely as descriptive benchmark anchors and are not interpreted as estimates of run-level central tendency or as evidence of statistical superiority. Cross-study differences are therefore discussed only after considering compatibility in objective definition, benchmark setting, and feasibility constraints.

Table 4: Selected best-found passenger-cost values (min) and historical benchmark comparators.

Instance	Present study	Mumford [?]	Nayeem [?]	Kılıç [?]
Mandl	10.12	10.45	9.86	10.21
Mumford0	21.34	22.18	21.89	22.45
Mumford1	28.76	30.12	29.34	31.02
Mumford2	35.89	38.45	37.12	—
Mumford3	42.31	44.28	43.56	—
Nottingham100	41.23	43.67	—	—
Edinburgh200	52.18	55.34	—	—

Note. Values in this table represent best-found passenger-cost objectives and are retained for descriptive historical comparison only. They should not be interpreted as run-level averages or as evidence of statistical superiority across algorithms. Bold entries indicate the lowest value in each row.

4.7.2 Statistical and Computational Interpretation

Hypervolume and spacing are retained as formally defined Pareto-set performance indicators under the instance-specific evaluation convention described in Sections ?? and ?. Because complete matched run-level indicator records are not available across Configurations A–D, these metrics are not used for inferential statistical comparison in the present results. Duplicate elimination is implemented as a deterministic population-hygiene step within Configuration D rather than as a separate metaheuristic. A D-versus-no-filter ablation is therefore not reported, because such an analysis would require paired runs performed under identical conditions with duplicate filtering as the only changed component.

Convergence is assessed within the fixed 200-generation computational budget specified in Section ?. Because objective evaluation is dominated by the $\mathcal{O}(n_T^3)$ all-pairs shortest-path computation on the transfer-expanded graph, larger networks are expected to consume disproportionately more computation even when N and $G_{\max}$ are fixed. The paper therefore does not use wall-clock time as evidence of algorithmic superiority unless hardware, implementation language, parallelization, and stopping conditions are matched across methods.

The implementation analyzed in this study does not include GPU acceleration, and no hardware-specific speedup is claimed. Practical scalability is therefore interpreted from the benchmark sizes and the explicit complexity model rather than from an undocumented runtime figure. Future runtime studies should report the processor/GPU model, thread count, software stack, evaluation count, and per-instance profiling so that computational efficiency can be compared reproducibly with parallel methods such as that of Zervas et al. [?].

4.8 Genetic Operator Analysis

The route operators serve complementary search roles. Add/Del operations provide local structural refinement; Exchange and Merge enable larger route reconfiguration; Remove-Overlapping suppresses redundant structure; 2-opt refines within-route ordering; and Invert-Exchange promotes diversification across route pairs. Because event-level operator logs are not available, no operator-specific dominance-rate percentages are reported; the interpretation is therefore mechanistic and linked to the established UTNDP operator literature [?, ?, ?].

4.9 Comparison with Selected Literature

Historical passenger-cost values in Table ?? (visualized in Figure ??) are complemented by a contemporary methodological comparison with directly related 2023–2026 peer-reviewed studies. Numerical comparison would be misleading when the decision space, objective definitions, transfer treatment, operator-cost indicator, or computational architecture differs. This distinction is essential: the present study treats recent studies as state-of-the-art comparators for positioning and methodology, while it limits numerical claims to genuinely commensurate objective definitions.

The selected studies illustrate the methodological diversity of recent transit-network optimization research. Some emphasize parallel or swarm-based search, others incorporate learned route-improvement policies, and others extend the decision space to frequency, energy, or charging decisions. These differences affect objective definitions, computational architecture, transfer treatment, and the interpretation of operator cost. For this reason, the present study uses these works primarily for methodological positioning and restricts direct numerical comparison to cases in which the underlying benchmark and objective definitions are sufficiently aligned.

From an algorithmic perspective, Hüsselmann et al. [?], Zervas et al. [?], and Kourepinis et al. [?] represent complementary directions in contemporary UTRP search. Hüsselmann et al. emphasize structured initialization and hyperheuristic operator management, Zervas et al. focus on parallel artificial-bee-colony search for larger instances, and Kourepinis et al. employ a discrete swarm-based strategy with attention to computational cost. These studies are methodologically relevant to the present framework, although differences in search architecture and operator-cost formulation limit direct numerical comparison.

A second group of studies extends the decision space or the learning mechanism beyond the present route-only formulation. Holliday et al. [?] employ deep reinforcement learning and graph neural networks for learned route improvement, whereas Martinez-Quezada et al. [?] and Yoo et al. [?] incorporate additional decisions such as service frequency, charging, or route

number. Consequently, these works are used here to position the methodological scope of the proposed framework rather than as directly commensurate numerical benchmarks.

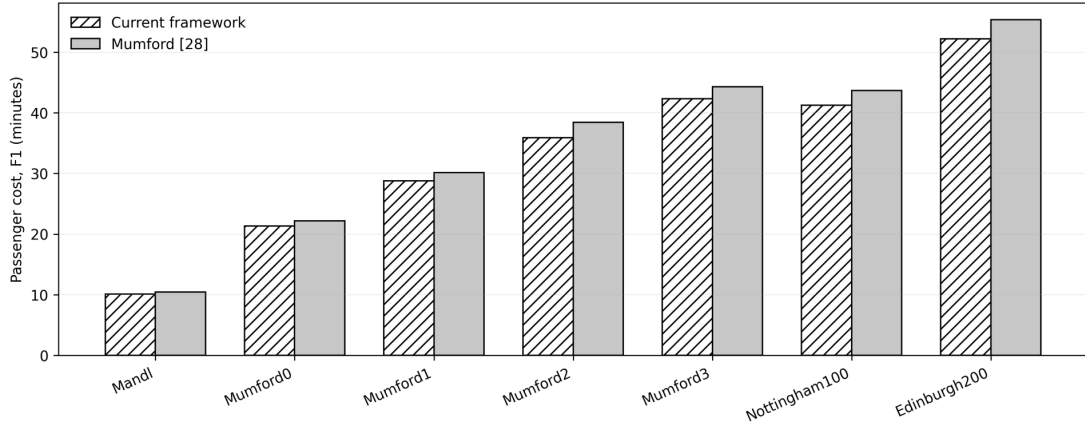


Figure 2: Historical passenger-cost comparison between the present framework and the results of Mumford [?] for the values reported in Table ??; this figure is not a state-of-the-art ranking against recent methods.

4.10 Summary

This section presented a demand-aware heuristic-initialized NSGA-II framework for the UTNDP combining

- a demand-aware heuristic initialization procedure that combines normalized travel-time and demand information;
- the NSGA-II framework [?] for Pareto ranking, elitist environmental selection, and crowding-distance preservation;
- a combined set of route-level mutation operators, with inherited components explicitly attributed to prior UTNDP and local-search literature [?, ?, ?];
- an edge-based Sørensen/Dice diversity measure with deterministic duplicate filtering as a population-management safeguard.

Table ?? provides descriptive passenger-cost comparisons across the benchmark set. Hypervolume and spacing remain formally defined Pareto-set performance indicators within the methodological framework, but the present results do not use incomplete run-level indicator records to support inferential or statistical-superiority claims. Component-wise performance attribution to initialization, selection, duplicate filtering, or parameter choice is likewise reserved for explicitly designed ablation experiments.

The combined operator set is intended to balance local refinement and structural diversification. In the absence of event-level operator logs, no numerical dominance-rate claim is made for an individual operator; the interpretation is limited to the structural function of each operator within the search process.

The benchmark evidence is interpreted as competitiveness under the explicitly stated route-only formulation, not as a universal best-known claim. Recent methods [?, ?, ?, ?, ?, ?] differ in operator-cost definitions, learned versus non-learned search, parallel architecture, or joint frequency/energy decisions. A defensible state-of-the-art claim would require identical benchmark data, transfer penalties, route-count constraints, objective definitions, and stochastic-run reporting, as well as matched hardware and stopping criteria where runtime is compared.

5 Conclusion

This study formulated the NP-hard Urban Transit Network Design Problem as a bi-objective route-set optimization problem and solved it with a demand-aware heuristic-initialized NSGA-II framework. The model minimizes demand-weighted passenger travel time and operator route-weight cost subject to coverage, route-length, transfer-connectivity, global-connectivity, and route-cardinality constraints. The computational architecture combines demand-aware initialization, feasibility-first repair or rejection, problem-specific route operators, Pareto ranking, crowding-distance preservation, and deterministic duplicate control.

Across the benchmark suite, the framework yields competitive passenger-cost values, including 21.34 min on Mumford0 and 52.18 min on Edinburgh200 in the descriptive comparisons of Table ???. Hypervolume and spacing remain formally defined Pareto-set performance indicators within the evaluation framework, but the present conclusions are restricted to numerical benchmark results directly supported by the available computational records; no run-level inferential or statistical-superiority claim is made. Duplicate filtering is treated as an internal population-management safeguard rather than as a separately validated algorithm. Frequency, vehicle capacity, and GPU acceleration are not part of the optimized decision space or implementation reported here.

The interpretation of the results is bounded by several explicit limitations. First, the infrastructure is fixed: stop relocation, stop addition/removal, and road-network redesign are outside the decision space. Second, OD demand is static and represented by a single planning-period matrix, so time-of-day demand variation, stochastic travel times, and disruption response are not modeled. Third, service frequency, fleet capacity, and vehicle energy constraints are exogenous; the model is therefore a route-design problem rather than a joint network-design-and-frequency problem. Fourth, the controlled regime $N = 200$ and $G_{\max} = 200$ is not the output

of an automated hyperparameter-optimization study, so the reported conclusions apply to this common computational budget. Fifth, Floyd–Warshall evaluation on the transfer-expanded graph creates an $\mathcal{O}(n_T^3)$ scalability bottleneck, and the present study does not claim hardware-independent wall-clock superiority over recent parallel or learned methods.

Future work should therefore proceed in four concrete directions. (i) Dynamic OD matrices estimated from automated fare collection and other demand streams can support time-dependent or robust route redesign. (ii) Electric-fleet extensions should jointly represent route structure, charging-location/charging-time constraints, battery energy consumption, and service frequency, following the broader decision scope illustrated by recent work on the electric transit network design and frequency-setting problem [?]. (iii) The dominant shortest-path cost should be reduced through sparse APSP methods, reusable path caches, and topology-aware acceleration rather than relying on an unspecified hardware speedup. (iv) Learned operator-selection policies or graph-neural heuristics can replace uniform/random mutation choice and be compared against transparent non-neural baselines, connecting the present operator framework to recent deep-reinforcement-learning approaches [?].

Declarations

Availability of Supporting Data

Data are available for research purposes upon reasonable request to the corresponding author.

Funding

The authors received no specific funding for this research.

Conflict of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have influenced the work reported in this paper.

Authors' Contributions

Reza Rajabi: Conceptualization, methodology, mathematical formulation, development of the demand-aware heuristic initialization, algorithm implementation, software, simulation, formal analysis, investigation, visualization, and writing—original draft.

Farzin Modarres Khiabani: Conceptualization, methodology, supervision, validation, optimization-model formulation, interpretation of the computational results, project administration, and writing—review and editing. **Saba Sadeghi Gavgani:** Methodology, validation, formal analysis, investigation, benchmark-based evaluation, interpretation of

results, literature review, and writing–review and editing. All authors reviewed and approved the final version of the manuscript and agreed to be accountable for the accuracy and integrity of the work.

Artificial Intelligence Statement

Artificial intelligence (AI) tools were used solely for language editing and manuscript refinement, including grammatical correction, sentence restructuring for clarity, and improvement of phrasing and readability. The AI tools were not used to generate scientific ideas, design experiments, perform analyses, interpret results, or develop scientific conclusions. All scientific content, methodological decisions, analyses, interpretations, and conclusions were independently developed and verified by the authors, who retain full responsibility for the final manuscript.

Publisher's Note

The publisher remains neutral regarding jurisdictional claims in published maps and institutional affiliations.

References

- [1] Almasi, M. H., Oh, Y., Sadollah, A., Byon, Y. J., Kang, S. (2021). “Urban transit network optimization under variable demand with single and multi-objective approaches using metaheuristics: The case of Daejeon, Korea”. *International Journal of Sustainable Transportation*, 15(5), 386–406. <https://doi.org/10.1080/15568318.2020.1821414>
- [2] Baker, B. M., Ayechev, M. A. (2003). “A genetic algorithm for the vehicle routing problem”. *Computers & Operations Research*, 30(5), 787–800. [https://doi.org/10.1016/S0305-0548\(02\)00051-5](https://doi.org/10.1016/S0305-0548(02)00051-5)
- [3] Baldacci, R., Christofides, N., Mingozzi, A. (2008). “An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts”. *Mathematical Programming*, 115(2), 351–385. <https://doi.org/10.1007/s10107-007-0178-5>
- [4] Beirão, G., Cabral, J. S. (2007). “Understanding attitudes towards public transport and private car: A qualitative study”. *Transport Policy*, 14(6), 478–489. <https://doi.org/10.1016/j.tranpol.2007.04.009>

- [5] Bell, J. E., McMullen, P. R. (2004). "Ant colony optimization techniques for the vehicle routing problem". *Advanced Engineering Informatics*, 18(1), 41–48. <https://doi.org/10.1016/j.aei.2004.07.001>
- [6] Brands, T., van Berkum, E. C. (2014). "Performance of a genetic algorithm for solving the multi-objective, multimodal transportation network design problem". *International Journal of Transportation*, 2(1), 1–20. <https://doi.org/10.14257/ijtt.2014.2.1.01>
- [7] Carvajal-Carreño, W., Cucala, A. P., Fernández-Cardador, A. (2014). "Optimal design of energy-efficient ATO CBTC driving for metro lines based on NSGA-II with fuzzy parameters". *Engineering Applications of Artificial Intelligence*, 36, 164–177. <https://doi.org/10.1016/j.engappai.2014.07.019>
- [8] Chai, S., Liang, Q. (2020). "An improved NSGA-II algorithm for transit network design and frequency setting problem". *Journal of Advanced Transportation*, 2020, 2895320. <https://doi.org/10.1155/2020/2895320>
- [9] Chakroborty, P., Dwivedi, T. (2002). "Optimal route network design for transit systems using genetic algorithms". *Engineering Optimization*, 34(1), 83–100. <https://doi.org/10.1080/03052150210909>
- [10] Chen, B., Zhang, R., Long, S., Sakdanuphab, R. (2024). "A multi-objective multi-period low-carbon location-routing problem: Improved NSGA-II approach". *IEEE Access*, 12, 51590–51605. <https://doi.org/10.1109/ACCESS.2024.3386584>
- [11] Clarke, G., Wright, J. W. (1964). "Scheduling of vehicles from a central depot to a number of delivery points". *Operations Research*, 12(4), 568–581. <https://doi.org/10.1287/opre.12.4.568>
- [12] Cooper, I. M., John, M. P., Lewis, R., Mumford, C. L., Olden, A. (2014). "Optimising large scale public transport network design problems using mixed-mode parallel multi-objective evolutionary algorithms". *2014 IEEE Congress on Evolutionary Computation (CEC)*, 2841–2848. <https://doi.org/10.1109/CEC.2014.6900362>
- [13] Croes, G. A. (1958). "A method for solving traveling-salesman problems". *Operations Research*, 6(6), 791–812. <https://doi.org/10.1287/opre.6.6.791>
- [14] Czech, Z. J., Czarnas, P. (2002). "Parallel simulated annealing for the vehicle routing problem with time windows". *Proceedings of the 10th Euromicro Workshop on Parallel, Distributed and Network-Based Processing*. <https://doi.org/10.1109/EMPDP.2002.994313>

- [15] Deb, K., Pratap, A., Agarwal, S., Meyarivan, T. (2002). "A fast and elitist multiobjective genetic algorithm: NSGA-II". *IEEE Transactions on Evolutionary Computation*, 6(2), 182–197. <https://doi.org/10.1109/4235.996017>
- [16] Faroqi, H. (2024). "Multiobjective route finding in a multimode transportation network by NSGA-II". *Journal of Engineering and Applied Science*, 71, 81. <https://doi.org/10.1186/s44147-024-00417-7>
- [17] Gardner, B., Abraham, C. (2007). "What drives car use? A grounded theory analysis of commuters' reasons for driving". *Transportation Research Part F: Traffic Psychology and Behaviour*, 10(3), 187–200. <https://doi.org/10.1016/j.trf.2006.09.004>
- [18] Gillett, B. E., Miller, L. R. (1974). "A heuristic algorithm for the vehicle-dispatch problem". *Operations Research*, 22(2), 340–349. <https://doi.org/10.1287/opre.22.2.340>
- [19] Guihaire, V., Hao, J.-K. (2008). "Transit network design and scheduling: A global review". *Transportation Research Part A: Policy and Practice*, 42(10), 1251–1273. <https://doi.org/10.1016/j.tra.2008.03.011>
- [20] Holliday, A., El-Geneidy, A., Dudek, G. (2025). "Learning heuristics for transit network design and improvement with deep reinforcement learning". *Transportmetrica B: Transport Dynamics*, 13(1), 2561863. <https://doi.org/10.1080/21680566.2025.2561863>
- [21] Hüsselmann, G., van Vuuren, J. H., Andersen, S. J. (2024). "An improved solution methodology for the urban transit routing problem". *Computers & Operations Research*, 163, 106481. <https://doi.org/10.1016/j.cor.2023.106481>
- [22] Jha, S. B., Jha, J. K., Tiwari, M. K. (2019). "A multi-objective meta-heuristic approach for transit network design and frequency setting problem in a bus transit system". *Computers & Industrial Engineering*, 130, 166–186. <https://doi.org/10.1016/j.cie.2019.02.025>
- [23] John, M. P., Mumford, C. L., Lewis, R. (2014). "An improved multi-objective algorithm for the urban transit routing problem". *Evolutionary Computation in Combinatorial Optimisation, Lecture Notes in Computer Science*, 8600, 49–60. https://doi.org/10.1007/978-3-662-44320-0_5
- [24] Kılıç, F., Gök, M. (2014). "A demand based route generation algorithm for public transit network design". *Computers & Operations Research*, 51, 21–29. <https://doi.org/10.1016/j.cor.2014.05.001>

- [25] Kourepinis, V., Iliopoulou, C., Tassopoulos, I. X., Beligiannis, G. N. (2024). “An artificial fish swarm optimization algorithm for the urban transit routing problem”. *Applied Soft Computing*, 155, 111446. <https://doi.org/10.1016/j.asoc.2024.111446>
- [26] Li, L., Zhang, C., Niu, C., Zhang, H. (2025). “Parametric multi-objective optimization of urban block morphology using NSGA-II: A case study in Wuhan, China”. *Sustainability*, 17(21), 9724. <https://doi.org/10.3390/su17219724>
- [27] Mandl, C. E. (1980). “Evaluation and optimization of urban public transportation networks”. *European Journal of Operational Research*, 5(6), 396–404. [https://doi.org/10.1016/0377-2217\(80\)90126-5](https://doi.org/10.1016/0377-2217(80)90126-5)
- [28] Martinez-Quezada, D. O., Cortés, C. E., Mauttone, A., Munizaga, M. A. (2026). “A multi-objective optimization approach for the electric transit network design and frequency setting problem”. *Applied Mathematical Modelling*, 154, 116669. <https://doi.org/10.1016/j.apm.2025.116669>
- [29] Miandoabchi, E., Farahani, R. Z., Dullaert, W., Szeto, W. Y. (2012). “Hybrid evolutionary metaheuristics for concurrent multi-objective design of urban road and public transit networks”. *Networks and Spatial Economics*, 12(3), 441–480. <https://doi.org/10.1007/s11067-011-9163-x>
- [30] Momenitabar, M., Mattson, J. (2021). “A multi-objective meta-heuristic approach to improve the bus transit network: A case study of Fargo–Moorhead area”. *Sustainability*, 13(19), 10885. <https://doi.org/10.3390/su131910885>
- [31] Mumford, C. L. (2013). “New heuristic and evolutionary operators for the multi-objective urban transit routing problem”. *2013 IEEE Congress on Evolutionary Computation*, 939–946. <https://doi.org/10.1109/CEC.2013.6557668>
- [32] Nayeem, M. A., Rahman, M. K., Rahman, M. S. (2014). “Transit network design by genetic algorithm with elitism”. *Transportation Research Part C: Emerging Technologies*, 46, 30–45. <https://doi.org/10.1016/j.trc.2014.05.002>
- [33] Nikolić, M., Teodorović, D. (2013). “Transit network design by Bee Colony Optimization”. *Expert Systems with Applications*, 40(15), 5945–5955. <https://doi.org/10.1016/j.eswa.2013.05.002>
- [34] Qi, L., Zhang, R., Luan, W., Li, M., Guo, X. (2025). “Multi-objective optimization for multi-modal route planning integrating shared taxi and bus”. *Computing and Informatics*, 44(4), 769–799. https://doi.org/10.31577/cai_2025_4_769

- [35] Seifpour, M., Asghari, S. A., Ghobaei-Arani, M. (2024). “A stochastic multi-objective optimization method for railways scheduling: A NSGA-II-based hybrid approach”. *The Journal of Supercomputing*, 80(2), 2128–2163. <https://doi.org/10.1007/s11227-023-05529-0>
- [36] Stradling, S. G., Carreno, M., Rye, T., Noble, A. (2007). “Passenger perceptions and the ideal urban bus journey experience”. *Transport Policy*, 14(4), 283–292. <https://doi.org/10.1016/j.tranpol.2007.02.003>
- [37] Tan, K. C., Chew, Y., Lee, L. H. (2006). “A hybrid multi-objective evolutionary algorithm for solving truck and trailer vehicle routing problems”. *European Journal of Operational Research*, 172(3), 855–885. <https://doi.org/10.1016/j.ejor.2004.11.019>
- [38] Tan, K. C., Chew, Y. H., Lee, L. H. (2006). “A hybrid multiobjective evolutionary algorithm for solving vehicle routing problem with time windows”. *Computational Optimization and Applications*, 34(1), 115–151. <https://doi.org/10.1007/s10589-005-3070-3>
- [39] Toth, P., Vigo, D. (2003). “The granular tabu search and its application to the vehicle-routing problem”. *INFORMS Journal on Computing*, 15(4), 333–346. <https://doi.org/10.1287/ijoc.15.4.333.24890>
- [40] Yoo, S., Lee, J. B., Han, H. (2023). “A reinforcement learning approach for bus network design and frequency setting optimisation”. *Public Transport*, 15, 503–534. <https://doi.org/10.1007/s12469-022-00319-y>
- [41] Zervas, A., Iliopoulou, C., Tassopoulos, I. X., Beligiannis, G. N. (2024). “Solving large-scale instances of the urban transit routing problem with a parallel artificial bee colony-hill climbing optimization algorithm”. *Applied Soft Computing*, 167, 112335. <https://doi.org/10.1016/j.asoc.2024.112335>
- [42] Zhang, Z., Cheng, X., Xing, Z., Gui, X. (2023). “Pareto multi-objective optimization of metro train energy-saving operation using improved NSGA-II algorithms”. *Chaos, Solitons & Fractals*, 176, 114183. <https://doi.org/10.1016/j.chaos.2023.114183>
- [43] Zitzler, E., Thiele, L. (1998). “Multiobjective optimization using evolutionary algorithms—A comparative case study”. *Parallel Problem Solving from Nature — PPSN V, Lecture Notes in Computer Science*, 292–301. <https://doi.org/10.1007/BFb0056872>

Authors' Bio-sketches

Reza Rajabi is an academic and researcher with approximately 20 years of teaching and academic experience at Islamic Azad University, Sarab Branch, Iran. He is currently a faculty member at Islamic Azad University, Sarab Branch, and a Ph.D. candidate at Islamic Azad University, Tabriz Branch. His academic activities include university teaching, research, and participation in scholarly projects related to his field of specialization.

Farzin Modarres Khiabani is an Associate Professor and faculty member at Islamic Azad University, Tabriz Branch, Iran, with approximately 21 years of academic, research, and teaching experience. His academic background includes extensive involvement in university education, scholarly research, and the supervision and development of research activities related to his area of specialization. Corresponding author. Email: f.modarres@iaui.ir

Saba Sadeghi Gavgani is an Assistant Professor and faculty member at Islamic Azad University, Sarab Branch, Iran, with approximately 16 years of academic and teaching experience. She has extensive experience in university education, academic research, and scholarly activities, and has contributed to the development and implementation of research projects within her field of expertise.