

A Novel Approach to Route Similarity Measures for Shared Mobility Matching Systems

Antoni Negrea

2025

Abstract

Shared mobility systems rely on accurately identifying trips that can be efficiently and fairly matched. A central challenge in such systems is the definition of route similarity measures that are both computationally efficient and effective in capturing meaningful spatial overlap between user trajectories.

This paper addresses the problem of route similarity evaluation for shared mobility matching, with a focus on simplicity, interpretability, and reproducibility. The main contribution is a lightweight similarity model that combines a geometric proximity measure with a segment overlap metric into a tunable linear similarity function. Unlike many existing approaches that rely on complex spatial clustering or black-box optimization, the proposed method is intentionally simple and analytically transparent.

The paper further introduces and compares two matching strategies: a static global optimization approach based on partition merging, and a dynamic greedy assignment strategy suitable for real-time scenarios. Experimental validation is conducted on both synthetic datasets and prepared real-world trajectory datasets. The results demonstrate that the proposed similarity model achieves competitive travel efficiency, while offering explicit control over fairness through parameter tuning.

Overall, the study shows that simple similarity measures, when carefully combined and validated, can provide effective solutions for shared mobility matching problems, while remaining easy to implement and extend.

Contents

Research Classification	3
1 Introduction	4
1.1 Background and Motivation	4
1.2 Problem Statement	4
1.3 Related Work and Open Challenges	5
1.4 Original Contributions	5
1.5 Research Questions	6
1.6 Paper Structure	6
2 Modeling the Experimental Part	7
2.1 The Dataset	7
2.1.1 Representation of Urban Routes	7
2.1.2 User Profiles	8
2.2 Simplified Similarity Measures	8
2.2.1 Geometric Similarity	8
2.2.2 Segment Overlap	8
2.2.3 The Final Similarity Function	8
2.3 Matching Algorithms	8
2.3.1 Static Matching — Partition Merging	8
2.3.2 Dynamic Matching — Greedy	9
2.4 Proposed Experiments	9
2.4.1 Experiment 1: Static vs. Greedy	9
2.4.2 Experiment 2: Impact of Parameters α, β	9
2.4.3 Experiment 3: Scalability	9
2.5 Validation Methods	9
2.5.1 Internal Validation	10
2.5.2 External Validation	10

2.6	Conclusion	10
3	Case Study on the Initial Dataset	11
3.1	Dataset Description	11
3.2	Experimental Code Implementation	11
3.3	Results and Analysis	12
3.3.1	Experiment 1: Static vs. Greedy ($\alpha = \beta = 0.5$)	12
3.3.2	Experiment 2: Impact of Parameters (α, β) on 'Partial' Scenario	13
3.4	Conclusion	14
4	Preparation for Validation on Real Data	15
4.1	Selecting Datasets and Preprocessing	15
4.2	Existing Work	16
4.3	Comparison of the Proposed Approach	16
4.4	Metrics for Validation	17
4.5	Conclusion	18

Research Classification

ACM Computing Classification System (CCS)

- Information systems → Geographic information systems
- Information systems → Location based services
- Computing methodologies → Optimization algorithms
- Applied computing → Transportation

AMS Subject Classification

- 90B06 – Transportation, logistics
- 68W15 – Parallel and distributed algorithms (for matching strategies)
- 65K10 – Optimization and variational techniques

Chapter 1

Introduction

Urban transportation systems face increasing pressure due to traffic congestion, environmental concerns, and inefficient use of vehicle capacity. Shared mobility and carpooling systems aim to mitigate these issues by enabling multiple travelers with similar routes to share the same vehicle. A core challenge in such systems is determining when two or more trips are “similar enough” to be matched without causing excessive detours or unfair delays.

1.1 Background and Motivation

Most shared mobility platforms rely on route similarity measures to decide which trips can be merged. Existing approaches often employ complex spatial clustering, probabilistic models, or machine-learning-based similarity estimators. While powerful, these methods tend to be computationally expensive, difficult to tune, and hard to reproduce across datasets and cities.

In practice, many real-world systems require fast, interpretable, and easily adjustable matching algorithms, particularly in dynamic, real-time environments.

1.2 Problem Statement

Given a set of user trips, each represented as a spatial route with temporal constraints, the objective is to group compatible trips such that:

- the total travel distance is reduced (efficiency);

- individual detours are bounded (fairness);
- the matching process remains computationally feasible.

The problem addressed in this paper is the design of a route similarity model that balances simplicity and effectiveness, while enabling both static and dynamic matching strategies.

1.3 Related Work and Open Challenges

Previous studies have explored spatial density models, graph-based similarity, and time-space optimization techniques. However, several challenges remain insufficiently addressed:

- lack of interpretability of similarity scores;
- limited control over efficiency–fairness trade-offs;
- high computational complexity for large-scale systems.

1.4 Original Contributions

This paper makes the following original contributions:

- a simple yet expressive route similarity function combining geometric distance and segment overlap;
- a unified cost formulation applicable to both static and greedy matching;
- a detailed experimental validation illustrating the efficiency–equity trade-off;
- a reproducible framework suitable for extension to real-world datasets.

1.5 Research Questions

The study aims to answer the following research questions:

- Can simple similarity measures achieve competitive performance compared to more complex models?
- How does parameter tuning affect efficiency and fairness?
- What are the trade-offs between static and dynamic matching strategies?

1.6 Paper Structure

The remainder of the paper is organized as follows. Chapter 2 introduces the proposed similarity model and matching approach. Chapters 3 and 4 describe the experimental setup and controlled case study. Chapter 5 prepares the validation on real-world datasets. Finally, conclusions and future research directions are discussed.

Chapter 2

Modeling the Experimental Part

This chapter rigorously describes the data used, the planned experiments, the mathematical modeling of the similarity measures, the algorithms compared, and the validation methods. The goal is to demonstrably prove, in a reproducible and analytically supported manner, that the proposed approach brings improvements over existing methods found in the literature.

2.1 The Dataset

2.1.1 Representation of Urban Routes

Each route is modeled as an ordered list of GPS points:

$$R = \{(lat_1, lon_1), \dots, (lat_n, lon_n)\}.$$

The data sources are:

- simplified urban road network
- artificially simulated routes for controlled scenarios

To reduce complexity, the GPS points are projected onto a discretized network $G = (V, E)$, where V are intersections and E are segments.

2.1.2 User Profiles

Each user is associated with a triplet:

$$U_i = (o_i, d_i, t_i),$$

where o_i is the origin, d_i the destination, and t_i the temporal interval (time window).

2.2 Simplified Similarity Measures

The proposed methodology uses two measures that are easy to implement:

2.2.1 Geometric Similarity

For two routes R_1, R_2 :

$$S_{geo}(R_1, R_2) = 1 - \frac{1}{|R_1|} \sum_{p \in R_1} \min_{q \in R_2} d(p, q),$$

where $d(p, q)$ is the Haversine distance.

2.2.2 Segment Overlap

$$S_{overlap}(R_1, R_2) = \frac{|R_1 \cap R_2|}{\max(|R_1|, |R_2|)}.$$

2.2.3 The Final Similarity Function

A simple linear function:

$$S_{final}(R_1, R_2) = \alpha S_{geo}(R_1, R_2) + \beta S_{overlap}(R_1, R_2),$$

where $\alpha + \beta = 1$.

2.3 Matching Algorithms

2.3.1 Static Matching — Partition Merging

The objective is to group users such that the cost is minimized:

$$\text{cost}(G) = \sum_{i, j \in G} (1 - S_{final}(R_i, R_j)).$$

2.3.2 Dynamic Matching — Greedy

For a new request:

$$\Delta\text{cost} = \text{cost}(G \cup \{U_k\}) - \text{cost}(G).$$

The request is allocated to the group with the minimum Δcost .

2.4 Proposed Experiments

2.4.1 Experiment 1: Static vs. Greedy

The comparison between the two algorithms (Static Partition Merging and Dynamic Greedy) uses two principal metrics:

1. **Efficiency** (Travel Gain): The reduction in total travel distance.

$$\text{TravelGain} = \frac{D_{solo} - D_{shared}}{D_{solo}}.$$

2. **Equity** (Maximum Relative Detour, MRD): The highest proportional increase in travel distance/time experienced by any single rider in a shared group. This assesses the fairness of the solution.

2.4.2 Experiment 2: Impact of Parameters α, β

The influence of weights on matching quality is analyzed using both the **Travel Gain** and **MRD** metrics to assess the trade-off between efficiency and equity.

2.4.3 Experiment 3: Scalability

We measure:

- runtime;
- memory used.

2.5 Validation Methods

Validation is exclusively numerical:

2.5.1 Internal Validation

Repeated simulations with artificially generated data.

2.5.2 External Validation

Comparison of results with:

- Xia & Curtin (2019) – spatial model;
- Duan (2018) – partition merging;
- Sun (2023) – greedy.

2.6 Conclusion

The experimental model is simplified, reproducible, and easy to implement. It allows for the evaluation of similarity functions and matching algorithms.

Chapter 3

Case Study on the Initial Dataset

This chapter presents a controlled experiment performed on a small simulated dataset to validate the proposed methodology in a simple scenario.

3.1 Dataset Description

The initial set contains:

- 10 short routes with controlled characteristics (50–200 m);
- close or distant origins and destinations;
- simple intersections for ease of implementation.

Three types of scenarios are included, each run with 10 routes:

1. nearly identical routes (IDENTICAL);
2. partially overlapping routes (PARTIAL);
3. completely different routes (DIFFERENT).

3.2 Experimental Code Implementation

The practical component consists of implementing the following in Python:

- a route generator;
- the S_{geo} and $S_{overlap}$ functions;
- the static and greedy algorithms;
- the metrics measurement module.

Code structure:

```
ExperimentalPart/
  route_generator.py
  similarity.py
  static_matching.py
  greedy_matching.py
  metrics.py
  main.py
```

3.3 Results and Analysis

The initial simulations were executed using $N = 10$ routes per scenario, comparing the Static Matching (Partition Merging) against the Dynamic Matching (Greedy) approach. The results are summarized below.

3.3.1 Experiment 1: Static vs. Greedy ($\alpha = \beta = 0.5$)

The initial experiment focuses on baseline performance using balanced similarity weights.

- **Identical Scenario:** As expected, both algorithms performed optimally, merging all 10 routes into a single group, yielding the maximum possible 90.00% Travel Gain and zero detour (0.00% MRD). This validates the algorithms' ability to identify perfect matches.
- **Partial Scenario:** The **Static Matching** algorithm achieved a slightly higher Travel Gain (18.87%) compared to the Greedy approach (15.09%), indicating that its global optimization view resulted in marginally more efficient overall groupings, even though both formed the same number of groups (6) and had the same average size (1.67) and MRD (50.00%).

Table 3.1: Experiment 1: Static vs. Greedy Comparison ($\alpha = 0.5, \beta = 0.5$)

Scenario	Algorithm	Groups	Avg. Size	Travel Gain	MRD
IDENTICAL	Static	1	10.00	90.00%	0.00%
	Greedy	1	10.00	90.00%	0.00%
PARTIAL	Static	6	1.67	18.87%	50.00%
	Greedy	6	1.67	15.09%	50.00%
DIFFERENT	Static	8	1.25	7.69%	100.00%
	Greedy	6	1.67	15.38%	100.00%

- **Different Scenario:** The results here are highly revealing. While both algorithms correctly showed low efficiency and high detours (100.00% MRD), the Greedy algorithm unexpectedly yielded a better Travel Gain (15.38% vs. Static’s 7.69%) despite forming fewer groups (6 vs. 8). This suggests that in scenarios with poor inherent matchability, the sequential nature of the Greedy algorithm might, by chance, establish a few highly efficient initial groups that the Static algorithm’s global, similarity-driven cost function failed to identify under this specific weight setting.

3.3.2 Experiment 2: Impact of Parameters (α, β) on ‘Partial’ Scenario

This experiment used the PARTIAL scenario as a testbed to analyze how the relative weighting of geometric similarity (α) versus segment overlap (β) affects efficiency and equity.

- **Geometric-Heavy ($\alpha = 0.9, \beta = 0.1$):** This setting proved to be the most successful for maximizing efficiency, with the **Greedy algorithm achieving the highest Travel Gain (30.91%)** across all tests. This result confirms that S_{geo} (geometric proximity) is the most informative metric in our similarity function. However, the Static algorithm achieved a high gain (29.09%) while maintaining a significantly lower detour (**MRD 20.00%** vs. Greedy’s 50.00%), highlighting the trade-off: Static matching offers superior equity for similar efficiency.

Table 3.2: Experiment 2: Parameter Impact on Matching Quality (PARTIAL Scenario)

α	β	Algorithm	Travel Gain	MRD
0.1 (Overlap-Heavy)	0.9	Static	25.45%	0.00%
		Greedy	25.45%	0.00%
0.5 (Balanced)	0.5	Static	29.09%	20.00%
		Greedy	20.00%	50.00%
0.9 (Geometric-Heavy)	0.1	Static	29.09%	20.00%
		Greedy	30.91%	50.00%

- **Overlap-Heavy** ($\alpha = 0.1, \beta = 0.9$): This setting resulted in perfect equity (0.00% MRD) for both algorithms, but at the cost of constrained efficiency (25.45% gain). This high β weight makes the similarity function overly strict, only permitting near-identical route matches, which limits the potential for efficiency gains by excluding feasible, but slightly detoured, matches.

3.4 Conclusion

The initial set **quantitatively confirms** that the two simple similarity measures are sufficient for relevant and measurable experiments. The results validate the model by demonstrating clear performance differences—for instance, the dependence on parameter weighting and the inherent trade-off between the Static (equity-focused) and Greedy (efficiency-focused) algorithms. The experiments successfully demonstrated the sensitivity and impact of the final similarity function’s weighting, with the S_{geo} component proving to be a critical factor for high-quality matching.

Chapter 4

Preparation for Validation on Real Data

This chapter establishes the technical requirements and analytical framework necessary for the external validation of the proposed similarity measures and matching algorithms. The final validation will be performed on real-world trajectory datasets, allowing for performance comparison against established approaches in the literature.

4.1 Selecting Datasets and Preprocessing

Validation will be performed on datasets frequently used in academic literature, specifically:

- **Porto Taxi Trajectory Dataset:** Provides a large volume of trips with fine-grained temporal and spatial data.
- **Beijing Trajectory Dataset:** Offers contrasting urban road network characteristics, often used for scalability tests.
- **Real OpenStreetMap (OSM) Maps:** Used as the underlying graph $G = (V, E)$ onto which all raw GPS data must be projected.

Data Acquisition and Map Matching

The transition from raw GPS data to the discrete route representation used by the model is non-trivial. The discrete network representation requires each

raw route $R_{raw} = \{(lat_1, lon_1), \dots\}$ to undergo a **Map Matching** process. This process converts the noisy GPS coordinates into an ordered sequence of segments (edges) on the OSM road network.

The final representation of a route R_i for validation purposes must be a sequence of road segment IDs:

$$R_i = \{s_1, s_2, \dots, s_m\}, \quad s_j \in E.$$

This segment-based representation is crucial, as it allows for the precise calculation of $\mathbf{S}_{\text{overlap}}$ and accurate accounting of $\mathbf{D}_{\text{shared}}$ for the Travel Gain metric.

4.2 Existing Work

The proposed approach is evaluated against three established methods, chosen for their distinct strategies in ride-sharing matching:

- **Xia & Curtin (2019) – Spatial Density Model:** This work utilizes complex spatial clustering algorithms. Our approach contrasts sharply by using a simplified, weighted combination (\mathbf{S}_{geo}) which prioritizes computational speed and tunability over complex density-based heuristics.
- **Duan (2018) – Partition Merging:** Duan’s method relies on maximizing trip efficiency. The comparison here will focus on their specific, often proprietary, cost function against our generalized cost function: $\text{cost}(G) = \sum(1 - S_{\text{final}})$. We aim to demonstrate that a simple, similarity-driven cost function is competitive with more tailored models.
- **Sun (2023) – Time-Constrained Greedy:** Sun’s approach uses a robust time-dependent cost function within a greedy framework. We compare our purely spatial $\mathbf{S}_{\text{final}}$ score to their time-space cost to isolate the impact of the similarity measures themselves on matching quality, particularly on the **Travel Gain** and **Fairness** metrics.

4.3 Comparison of the Proposed Approach

Our methodology offers two primary advantages over the existing work:

- **Tunable Similarity Function:** The linear combination $S_{final} = \alpha S_{geo} + \beta S_{overlap}$ provides explicit control over the influence of geometric proximity versus shared infrastructure. Experiments will systematically test $\alpha \in [0, 1]$ to identify the optimal balance, a level of control often obfuscated in black-box models.
- **Algorithmic Flexibility:** By providing robust implementations of both Static (optimal, slow) and Greedy (fast, heuristic) matching strategies, the model allows researchers to choose the trade-off between matching quality (high Travel Gain) and computational runtime (scalability), depending on the application context (e.g., pre-scheduling versus real-time dynamic dispatch).

4.4 Metrics for Validation

While the controlled case study focused on basic efficiency, validation on real data requires comprehensive metrics that address efficiency, equity, and computational performance.

- **Efficiency (E): Average Distance Saved / Travel Gain**

$$\text{TravelGain} = \frac{D_{solo} - D_{shared}}{D_{solo}}$$

This remains the primary metric for measuring the overall success of the carpooling strategy in reducing total mileage.

- **Equity (Q): Maximum Relative Detour (Fairness)** Fairness is assessed by the detour experienced by any single rider. The **Relative Detour** (D_i) for a traveler i is calculated as the increase in their travel time (or distance) compared to driving alone:

$$D_i = \frac{T_{shared,i} - T_{solo,i}}{T_{solo,i}}$$

The system’s **Fairness** is then defined as the maximum detour imposed on any matched traveler, known as the **Maximum Relative Detour (MRD)**:

$$\text{MRD} = \max_i(D_i)$$

The objective is to maximize Travel Gain while constraining the MRD to an acceptable limit (e.g., $\text{MRD} \leq 20\%$).

- **Performance (P): Runtime** The total time required for the matching process will be measured as a function of the number of requests (N) to evaluate the practical scalability of both the static and greedy algorithms.

4.5 Conclusion

This chapter successfully establishes the analytical context and metric framework required for the final validation. By formally defining the data requirements, the comparative works, and the critical Maximum Relative Detour (MRD) metric, the subsequent chapters can proceed with the rigorous external validation necessary to prove the value of the proposed similarity model.