

A Path Fitting Accuracy Improvement and Evaluation Method Based on Minimum Toll and Multi-dimensional Anomaly Detection Engine

Jian Li

Road Network Monitoring and Emergency Response Center

Yunfang Zhao

945711740@qq.com

Road Network Monitoring and Emergency Response Center

Ke Zhang

China Waterborne Transport Research Institute

Xin Gao

Road Network Monitoring and Emergency Response Center

Lexiang Mei

Road Network Monitoring and Emergency Response Center

Xiaoyu Guo

Road Network Monitoring and Emergency Response Center

Article

Keywords: path fitting, minimum toll, multi-dimensional anomaly detection, road network topology, dual-indicator quality evaluation model, expressway tolling

Posted Date: July 16th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-9925755/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Scientific Reports on August 18th, 2026.
See the published version at <https://doi.org/10.1038/s41598-026-66590-2>.

A Path Fitting Accuracy Improvement and Evaluation Method Based on Minimum Toll and Multi-dimensional Anomaly Detection Engine

Jian Li^{†1}, Yunfang Zhao^{†1*}, Ke Zhang², Xin Gao¹,
Lexiang Mei¹, Xiaoyu Guo¹

(1 Road Network Monitoring and Emergency Response Center ,Beijing,10029,China;
2 China Waterborne Transport Research Institute, Beijing 100088, China)

[†]These authors contributed equally to this work.

*Correspondence: Yunfang Zhao (945711740@qq.com)

Abstract: Aiming at the problems of inaccurate path fitting and “same road, different tolls” caused by missing or abnormal gantry data in expressway network tolling, this paper proposes an innovative two-stage path fitting framework. The first stage (baseline fitting) rapidly generates spatiotemporally reasonable paths based on the whole-network topology. The second stage (optimized fitting) deeply integrates the business principle of “minimum toll” with a parallelized multi-dimensional anomaly detection engine, which systematically defines six types of real-world anomalies and provides their mathematical detection models. In addition, this paper is the first to propose a dual-indicator quality evaluation model, providing a quantitative basis for the reliability of fitting results. Experiments based on real transaction data show that, compared with traditional shortest-path fitting and baseline methods, the proposed framework improves path fitting accuracy by approximately 2.1% and reduces the billing deviation rate by nearly 70%. The confidence indicator exhibits a strong negative correlation with actual fitting error (Pearson coefficient = -0.87), verifying the effectiveness of the evaluation model. Finally, case scenarios demonstrate the practical application of the two-stage framework and evaluation model. This work provides a reliable theoretical and practical solution for realizing accurate and fair expressway network tolling.

Keywords: path fitting; minimum toll; multi-dimensional anomaly detection; road network topology; dual-indicator quality evaluation model; expressway tolling

1 Introduction

1.1 Research Status at Home and Abroad

The popularization of the national expressway “one network” operation mode, ETC, and free-flow tolling technologies has shifted vehicle toll calculation from rough

estimation based on the shortest path to precise billing based on the actual driving path [1]. In this context, high-precision restoration of the complete vehicle travel trajectory (i.e., path fitting) from sparse toll gantry transaction sequences has become the core of online tolling systems, and its accuracy directly affects billing fairness and user trust [2, 3]. However, gantry data often suffer from missing records, duplication, disorder, and various noise due to equipment failures, signal interference, severe weather, and other factors, leading to fitted paths deviating from the real trajectory and causing tolling disputes over “same road, different tolls” [4]. Therefore, developing robust, accurate, and business-rule-compliant path fitting algorithms has become a research hotspot and urgent demand in the field of intelligent transportation and tolling.

At the academic research level, the path fitting problem is closely related to map matching and trajectory repair. Early studies mainly processed dense GPS trajectory points and mapped point sequences to the road network through geometric, topological, or probabilistic methods (such as Hidden Markov Models) [5, 6]. However, toll gantry data are characterized by extreme sparsity (often several kilometers between points) and strong topological constraints (vehicles must travel along preset road networks), making traditional methods suitable for dense data no longer fully applicable [7]. For path fitting with sparse infrastructure data, research has evolved into graph search-based methods. These methods abstract the road network as a directed graph, with gantries as nodes and connectivity relations as edges. When data gaps occur, algorithms such as Dijkstra [8] and A* [9] are used to search for connecting paths. Recent studies have attempted to incorporate historical trajectory statistics [10] or machine learning models [11] to predict path selection probability, but the core optimization objectives of these methods are mostly “shortest distance” or “fewest gantries,” which are inconsistent with the fundamental business principle of tolling scenarios—“charging the lowest feasible fee for the user’s actual travel path” [12, 13].

In terms of data quality governance, trajectory anomaly detection has been widely studied to identify general anomalies such as GPS drift, sudden stops, and speeding [14]. In tolling scenarios, anomalies are more specific, such as duplicate transactions caused by antenna re-reading, false triggering of reverse gantries, and “path knotting” caused by timestamp disorder [15, 16]. However, existing path fitting research often handles anomalies in a simple and fragmented manner, lacking a systematic framework to collaboratively process complex and coupled anomaly scenarios such as head/tail

noise points, mid-journey noise points, path knotting, overspeed driving, illegal U-turns, and mixed multiple trips [17, 18].

In terms of result credibility evaluation, existing work mostly outputs “black box” results, lacking quantitative indicators for the reliability of fitted paths. When the system makes a choice among ambiguous paths, it cannot inform operators of the certainty of the choice, resulting in a lack of objective and explainable adjudication basis in tolling disputes [19, 20].

1.2 Research Contributions

A review of domestic and international research in the past five years reveals three main shortcomings of current expressway path fitting methods: (1) The optimization objective is disconnected from core business rules, failing to take the “minimum toll” principle as the fundamental criterion for path arbitration; (2) The anomaly handling mechanism is fragmented, lacking collaborative detection and cleaning capabilities for multi-dimensional, complex real-world anomalies; (3) Result evaluation lacks quantitative standards, and the reliability of fitted paths is unknown, leading to poor system interpretability.

This paper aims to systematically address these gaps by proposing a brand-new two-stage path fitting and evaluation framework. The main contributions of this work are as follows:

(1) A two-stage methodology of “connectivity guarantee → business rule optimization” is proposed. The framework clearly decouples technical feasibility (obtaining a connected path) from business optimality (obtaining the fairest billing path). The first stage ensures the existence of a basic solution, and the second stage deeply integrates the minimum toll principle and refined anomaly handling, realizing for the first time the leap from geometric connectivity to business practicality.

(2) A parallelized multi-dimensional anomaly detection engine is designed. For the first time, mathematical modeling is performed for six types of complex anomalies in expressway tolling scenarios (head/tail noise points, mid-journey noise points, knotted paths, overspeed paths, illegal U-turns, and secondary trip mixing), and a parallel detection process is designed, significantly improving fitting robustness under complex data conditions.

(3) A dual-indicator fitting quality evaluation model is established. Two quantitative indicators—“confidence” and “raw en-route ratio”—are innovatively proposed, providing for the first time a computable and explainable evaluation standard

for the reliability of path fitting results, greatly enhancing system transparency and credibility.

(4) Comprehensive experimental verification and scenario applications are completed. Sufficient experiments based on massive real data verify the significant improvement of the framework in accuracy and fairness, and demonstrate its application examples in production systems, proving its practical value.

The structure of the rest of this paper is as follows: Chapter 2 details the research framework and methods, including offline preprocessing, two-stage online fitting, the anomaly detection engine, and the reliability evaluation model for fitting results, followed by experimental verification. Chapter 3 presents case demonstrations and looks forward to future work. Chapter 4 summarizes the full paper.

2 Research Framework and Methods

Ethics Statement: This study utilized large-scale, fully anonymized ETC transaction data. All methods were carried out in accordance with relevant guidelines and regulations. The study protocol was approved by the Institutional Review Board (or the relevant ethics committee) of Road Network Monitoring and Emergency Response Center. Informed consent was waived by the board due to the retrospective nature of the data and the absence of any interaction with human subjects.

The two-stage framework proposed in this paper consists of offline preprocessing and online fitting. Online fitting is explicitly divided into two serial stages: baseline fitting (Stage I) and optimized fitting (Stage II), as shown in Figure 1.

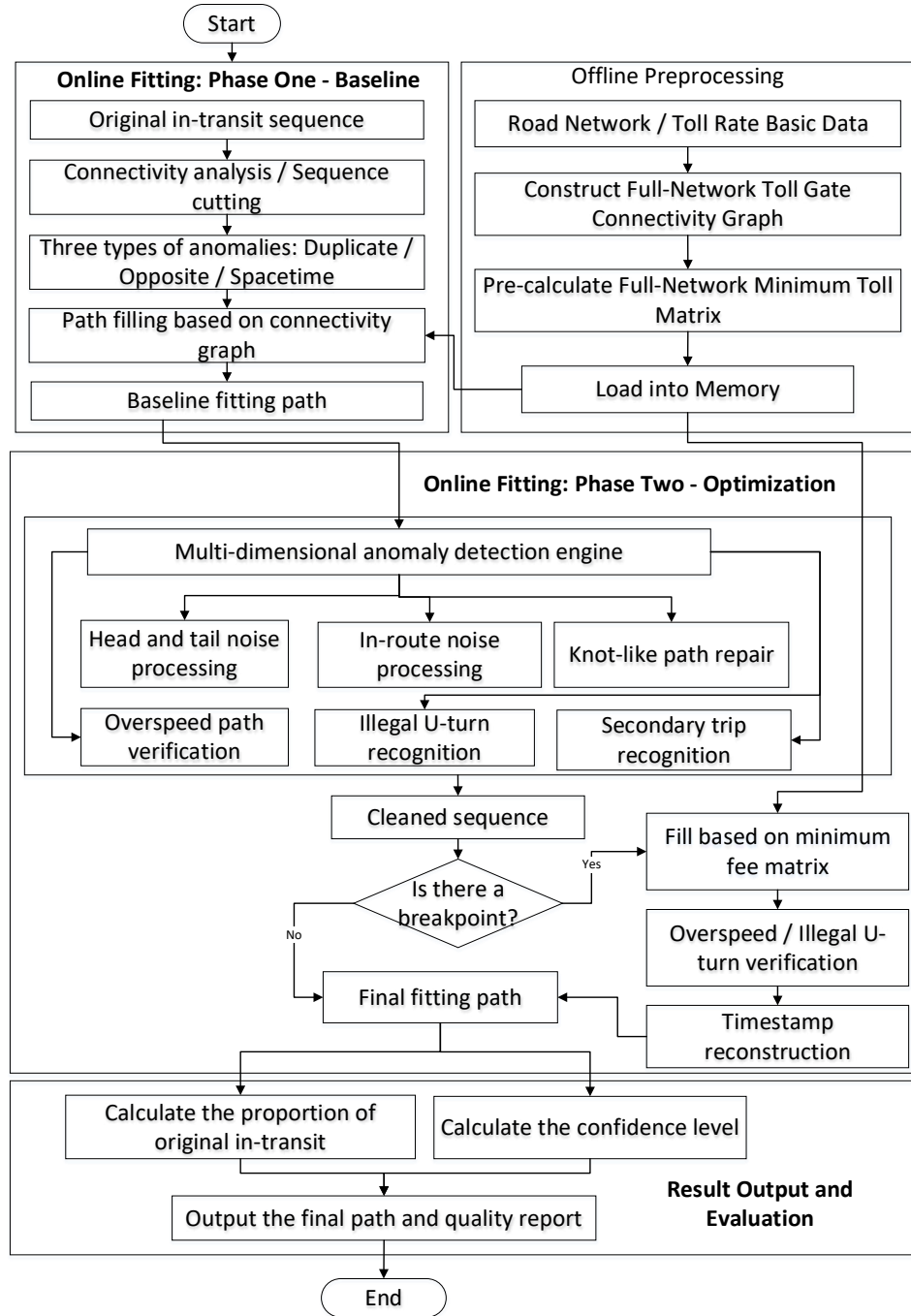


Figure 1. Overview of two-stage path fitting framework

2.1 Offline Preprocessing: Whole-Network Topology and Minimum Toll Matrix

2.1.1 Step 1: Construct a Gantry-Level Directed Connected Graph

Synchronize toll gantry, connectivity relation, and rate data daily. Take gantries as the node set V , and direct connectivity relations between gantries as the edge set E , to construct a directed graph $G = (V, E)$. V is the set of gantry vertices; E is the set of

connecting edges between gantries. For an edge e connecting gantries v_i and v_j , its initial weight $\omega_init(e)$ is the standard toll amount calculated based on mileage and rates.

2.1.2 Step 2: Precompute the Whole-Network Minimum Toll Matrix (Innovation Point)

To reduce the complexity of online path search from $O(E \log V)$ to $O(1)$, this paper introduces and precomputes a “whole-network minimum toll matrix” for the first time. Using $\omega_init(e)$ as the edge weight, $dist[v]$ represents the current minimum accumulated toll from the source point s to vertex v , and $prev[v]$ records the predecessor node of vertex v for subsequent path backtracking. An improved Dijkstra algorithm is used to calculate the minimum toll $cost_min(i, j)$ and corresponding path between any two nodes (v_i, v_j) in the graph. The core steps of the Dijkstra algorithm are as follows:

(1) Initialization

Before starting the calculation, the accumulated toll for all nodes except the source is set to infinity, and the predecessor node is set to null, as shown in formulas (1) and (2):

$$dist[v] = \begin{cases} 0 & \text{if } v=s \\ \infty & \text{if } v \neq s \end{cases} \quad (1)$$

$$prev[v] = null \quad \forall v \in V \quad (2)$$

(2) For each vertex u in graph G , traverse all its adjacent vertices v :

if $dist[v] > dist[u] + \omega(u, v)$ then $dist[v] = dist[u] + \omega(u, v)$ $prev[v] = u$

Where $dist[v]$ represents the current minimum toll from the source to vertex v , and $\omega(u, v)$ is the toll weight of edge (u, v) . $dist[u] + \omega(u, v)$ is the total toll of trying the new route. The minimum tolls for all node pairs are stored in matrix M_cost for instant query during the online stage.

(3) Greedy Strategy Selection

In each iteration, select the node with the smallest accumulated toll from the set of nodes whose shortest path has not yet been determined as the current u :

$$u = \arg \min_{v \in S} dist[v] \quad (3)$$

(4) Example of Minimum Toll

Figures 2, 3, and 4 show the minimum toll paths between any two gantry points (blue for start, red for end). Figure 2 has more waypoints (green points, 644 in total), Figure 3 has a moderate number (92), and Figure 4 has fewer (10).

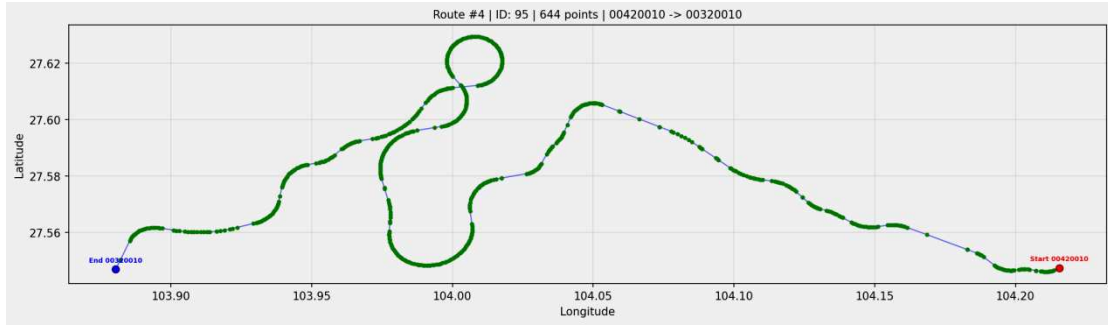


Figure 2. Connection relationship between any two gantries (with multiple points)

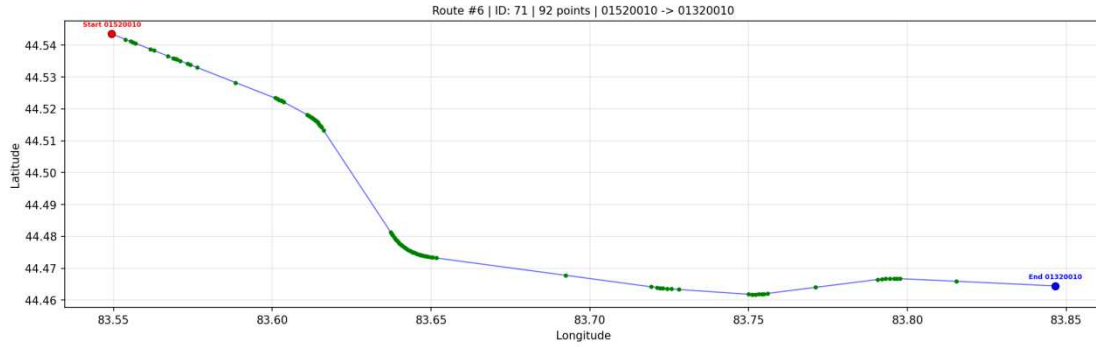


Figure 3. Connection relationship between any two gantries (moderate point position)

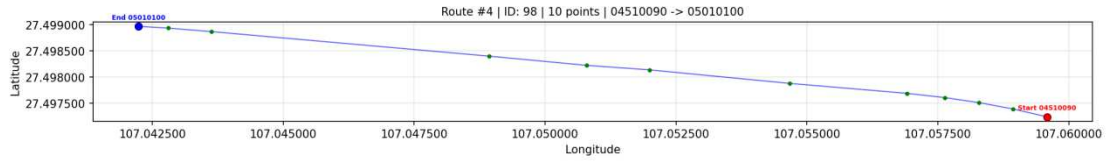


Figure 4. Connection relationship between any two gantries (with fewer points)

2.2 Online Fitting Stage I: Baseline Fitting (Connectivity Guarantee)

The goal of this stage is to quickly generate a spatially continuous and temporally basically reasonable path $P_{baseline}$ under the conditions of anomalies and missing data, serving as input for the optimization stage.

As shown in Figure 5, the schematic diagram of the expressway road network topology illustrates that the physical gantry nodes mentioned in this paper mainly refer to the blue points. The actual connectivity between the entrance toll station (red square) and exit toll station (orange square) is represented by the line connecting the blue points. The red line is the connection of points actually collected during vehicle travel. Green points are branch gantries, and gray squares are mid-journey toll stations. Path fitting is the process of finding the blue route based on the red points.

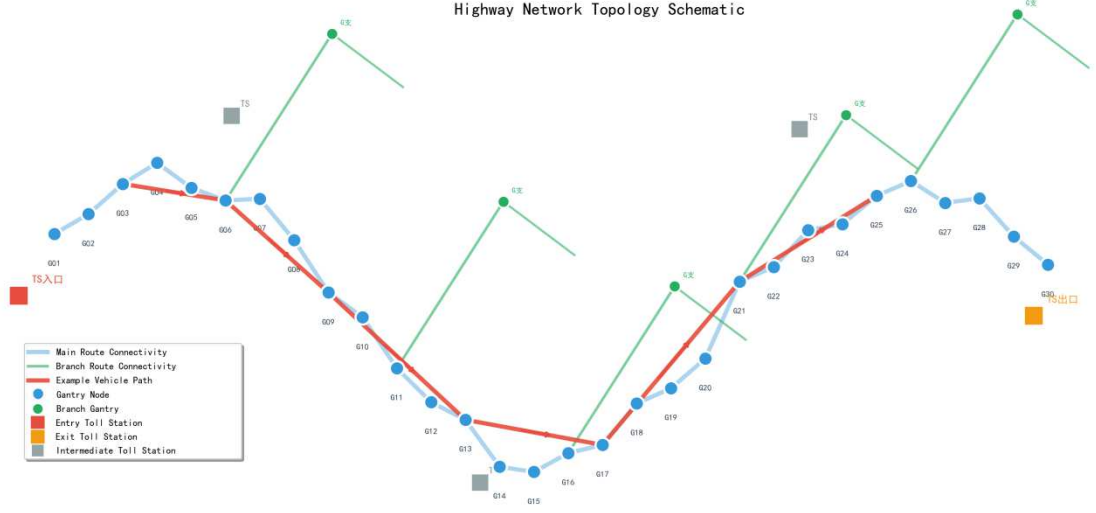


Figure 5. Schematic diagram of highway network topology

2.2.1 Sequence Cutting and Preliminary Cleaning

The input raw en-route gantry sequence is recorded as $S_{raw}=\{g_1, g_2, \dots, g_n\}$. First, cut the sequence based on the connected graph G : if consecutive gantries g_i and g_{i+1} are not connected, cut the sequence at g_i to obtain several internally connected subsequences Seg_k .

As shown in Figure 6, the complete trip is S1-A1-B1-C1-D1-E1-F1-G1-H1-G1. The actually queried points are A1, B1, D1, E1, G1, H1. The entrance toll station S1 can be inferred from the nearest gantry information according to the entrance logic. The segmentation results are as follows:

Segment 1: S1-A1-B1; Segment 2: D1-E1; Segment 3: G1-H1-S2.

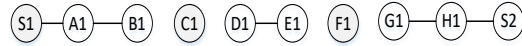


Figure 6. Schematic diagram of sequence segmentation

Subsequently, three types of basic cleaning are executed in parallel:

(1) **Duplicate Record Removal:** If $g_i == g_{i+1}$, remove g_{i+1} .

(2) **Reverse Transaction Identification:** For a suspected reverse gantry g_x , let its front and rear gantries be g_a and g_b . Calculate the hop count $H(a, \text{correct})$ from g_a to its correct gantry g_{correct} , and $H(\text{correct}, b)$ from g_{correct} to g_b ; similarly calculate $H(a, x)$ from g_a to g_x and $H(x, b)$ from g_x to g_b . If:

$$H(a, x) + H(x, b) > H(a, \text{correct}) + H(\text{correct}, b) \quad (5)$$

then g_x is determined to be a reverse gantry.

As shown in Figure 7, an example of reverse gantry identification: the queried en-route data is S1-A1-B1-C2-D2-E2-E1-F1-G1-S2, where D1 and D2 are a pair of reverse

gantries. Red points are erroneous, and gray C1 is a missing gantry point. According to antenna transaction judgment, there is a reverse transaction in this record. Let $U1$ be the number of gantry intervals between B1 and D1, $U2$ be the number between B1 and D2, $V1$ be the number between D1 and E1, and $V2$ be the number between D2 and E1. As shown, if $U2 + V2 > U1 + V1$, then D2 is confirmed as a reverse gantry, and the correct gantry should be D1. The correct result is: S1-A1-B1-C1-D1-E1-F1-S2.

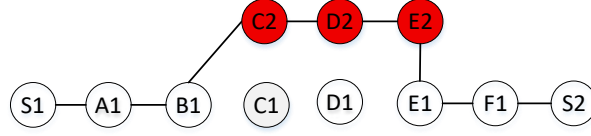


Figure 7. Schematic diagram of reverse gantry recognition

(3) Spatio-Temporal Anomaly Point Detection: For three consecutive points g_a, g_x, g_b , calculate the Euclidean distance D_{ab} between g_a and g_b , and the expected time T_{exp} for traveling at average speed V_{avg} :

$$T_{exp} = D_{ab} / V_{avg} \quad (6)$$

The actual time difference is:

$$T_{act} = t_b - t_c \quad (7)$$

If satisfied:

$$|T_{exp} - T_{act}| > \theta_{time} \vee T_{act} / T_{exp} > \theta_{ratio} \quad (\theta_{time}, \theta_{ratio} \text{ are threshold}) \quad (8)$$

then g_x is marked as a spatio-temporal anomaly point and removed.

As shown in Figure 8, an example of spatio-temporal anomaly point detection: $S1 \rightarrow A1 \rightarrow X1 \rightarrow B1 \rightarrow C1 \rightarrow S2$ is a path containing spatio-temporal anomaly points, where $X1$ is the anomaly point. Calculate the distance D of $A1-X1-B1$. Assuming an average speed of 80 km/h, the average travel time is $T = D/80$. Extract the travel times of $A1-X1$ and $X1-B1$ as t_i and t_{i+1} , and record the time difference $t_{i+1} - t_i$ as T' (the actual time passing through two toll units). If $T - T' > 3$ or $T'/T > 10$ or $T = 0$, it is considered a spatio-temporal anomaly point.

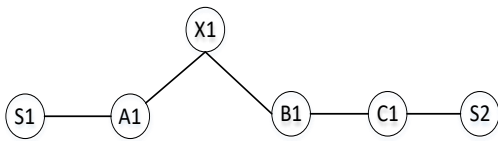


Figure 8. Examples of spatiotemporal outliers

2.2.2 Path Filling

For breakpoints(v_{start}, v_{end}) generated after cleaning, fill the path in graph G with the target of minimum hops. To balance efficiency and effect, a hybrid strategy is adopted:

(1) When the number of missing hops $n \leq N$ (e.g., $N = 17$), Depth-First Search (DFS) is used to quickly find a connected path. Core DFS code:

```
def dfs_iterative(graph, start):
    visited = set()
    stack = [start]
    while stack:
        node = stack.pop()
        if node not in visited:
            print(node)
            visited.add(node)
    stack.extend(reversed(graph[node])) # Maintain consistent order with recursion
```

(2) When $n > N$, the A-star Algorithm is used to improve search efficiency. The A-star Algorithm core steps include four modules: initialization, main loop condition, node selection (greedy strategy), and neighbor node update (relaxation operation), detailed as follows:

Definition: Let the directed graph of starting point S and ending point E be G , where N is a vertex in G representing a toll gate, and N_{min} specifically refers to the optimal node selected from *OpenList* in the current iteration. The priority queue used to store discovered but not yet evaluated nodes is *OpenList*, where nodes are sorted by $f(N)$ value during algorithm runtime. The collection of evaluated nodes is stored as a *CloseList*. The set of non passable nodes (unconnected gantries in road network topology) is stored as a *Blocklist*. $g(N)$ is the actual cost, which is the actual minimum cumulative toll from the starting point S to node N , such as $g(S)=0$. $H(N)$ is the heuristic function, which estimates the remaining toll from node N to endpoint E . In this paper, we use Euclidean distance:

$$h(N) = \sqrt{(x_{end} - x_n)^2 + (y_{end} - y_n)^2} \quad (8)$$

$f(N)$ is estimated total cost, i.e., the total cost estimate of node N , calculated as:

$$f(N) = g(N) + h(N) \quad (9)$$

The algorithm always prioritizes expanding the node with the smallest CX . $\omega(N_{min}, N)$ is the weight of the edge, and the toll of the edge from node N_{min} to its neighboring node N . In path fitting, this is the standard billing fee between two gantries. $parent(N)$ is the parent node pointer, which records the previous node on the current optimal path to node N . Used to backtrack and reconstruct the complete path from endpoint E after the search is completed. $Neighbors(N_{min})$ is the set of neighboring nodes, in graph G , the set of all nodes directly connected to node N_{min} with edges. f_{old} is the old estimated total cost, which is the $f(N)$ value of node N before the update. f_{new} is the new estimated cost, which is the new $f(N)$ value calculated when attempting to pass N_{min} , i.e.:

$$f_{new} = g(N_{min}) + \omega(N_{min}, N) + h(N) \quad (10)$$

Main Steps:

Step 1 Initialization: Define OpenList, CloseList, and initial node costs:

$$\begin{cases} OpenList \leftarrow \{S\} \\ CloseList \leftarrow \emptyset \\ g(S) \leftarrow 0, f(S) \leftarrow h(S) \end{cases} \quad (11)$$

Step 2 Main Loop Condition: Continue iterating while OpenList is not empty and the end point has not been found:

$$while\ OpenList \neq \emptyset \wedge goal \notin CloseList\ do\ h(S) \quad (12)$$

Step 3 Node Selection and Movement: Select the node N_{min} with the smallest $f(x)$ from OpenList and move it to CloseList:

$$\begin{cases} N_{min} = \operatorname{argmin}_{f(N), N \in OpenList} \\ CloseList \leftarrow CloseList \cup \{N_{min}\} \\ OpenList \leftarrow OpenList \setminus \{N_{min}\} \end{cases} \quad (13)$$

Step 4 Neighbor Node Update (Relaxation Operation): Traverse the neighbors N of N_{min} . If the path to N via N_{min} is better, update its parent node and cost:

$$for\ each\ N \in Neighbors(N_{min})\ do$$

$$P_baseline = \begin{cases} \text{if } N \in CloseList \vee N \in Blocklist: continue \\ \text{else if } N \notin OpenList: \\ \quad OpenList \leftarrow OpenList \cup \{N\} \\ \quad parent(N) \leftarrow N_{min} \\ \quad g(N) \leftarrow g(N_{min}) + w(N_{min}, N) \\ \quad f(N) \leftarrow g(N) + h(N) \\ \text{else:} \\ \quad fold \leftarrow g(N) + h(N) \\ \quad f_{new} \leftarrow g(N_{min}) + w(N_{min}, N) + h(N) \\ \quad \text{if } f_{new} < fold: \\ \quad \quad parent(N) \leftarrow N_{min} \\ \quad \quad g(N) \leftarrow g(N_{min}) + w(N_{min}, N) \\ \quad \quad f(N) \leftarrow f_{new} \end{cases} \quad (14)$$

To sum up, the baseline path filling formula is:

$$P_baseline = \begin{cases} dfs_iterative & n \leq 17 \\ f(n) = g(n) + h(n) & n > 17 \end{cases} n \in N + \quad (15)$$

2.3 Online Fitting Stage II: Optimized Fitting (Business Rule Optimization)

This stage takes $P_{baseline}$ input, performs deep cleaning and business-rule-based optimization. This part is the core innovation of this paper.

2.3.1 Parallelized Multi-dimensional Anomaly Detection Engine (Core Innovation Point)

This paper designs, for the first time, parallel detection modules for six typical expressway anomalies, launching six parallel anomaly detections on $P_baseline$. Each module is based on a strict mathematical model. The parallel multi-dimensional anomaly detection engine handles more complex scenarios.

(1) Head/Tail Noise Points: The nearest head and tail gantries (only 1 hop away from the toll station) are prone to being captured by reverse gantries near the station due to their proximity to opposite-direction gantries. Interference cannot be eliminated by connectivity alone. As shown in Figure 9, the actual queried passing order is B-E-A-C-F, where B is a noise point. Taking the entrance scenario as an example, the judgment flowchart is shown in Figure 10, with specific steps as follows:

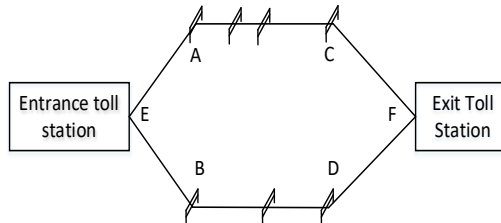


Figure 9. Schematic diagram of noise filtering scene at the beginning and end

1): Define the core variables. Let v_A be the speed at point A, v_B be the speed at point B, and v_{th} be the speed threshold, such as 130km/h. C_{EB} represents whether E and B are connected, with 1 being true and 0 being false. C_{EA} represents whether E and A are connected, with 1 being true and 0 being false. C_{EA_1} represents whether toll station E and A are connected within 1 hop, with 1 being true and 0 being false.

2):

Step 1: Initial screening of speed threshold to determine if A and B are speeding. Namely:

$$Step1 = \begin{cases} \text{if } v_A < v_{th} \vee v_B > v_{th} \rightarrow Step5 \\ \text{else if } v_A > v_{th} \vee v_B < v_{th} \rightarrow Step6 \\ \text{else if } v_A > v_{th} \wedge v_B > v_{th} \rightarrow Step5, Step6 \\ \text{else } v_A \leq v_{th} \wedge v_B \leq v_{th} \rightarrow Step2 \end{cases} \quad (16)$$

Step 2: Connectivity assessment between toll station E and B:

$$Step2 = \begin{cases} \text{if } C_{EB} = 1 \rightarrow Step3 \\ \text{else } C_{EB} = 0 \rightarrow Step5 \end{cases} \quad (17)$$

Step 3 Connectivity assessment between toll station E and A:

$$Step3 = \begin{cases} \text{if } C_{EA} = 1 \rightarrow Step4 \\ \text{else } C_{EA} = 0 \rightarrow Step6 \end{cases} \quad (18)$$

Step 4: Determination of intra hop connectivity between E and A:

$$Step4 = \begin{cases} \text{if } C_{EA_1} = 1 \rightarrow Step5 \\ \text{else if } C_{EA_1} = 0 \rightarrow Step6 \end{cases} \quad (19)$$

Step 5: Remove point B.

Step 6: Remove point A.

After fitting, use the connectivity relationship again to check if points A and B were mistakenly deleted.

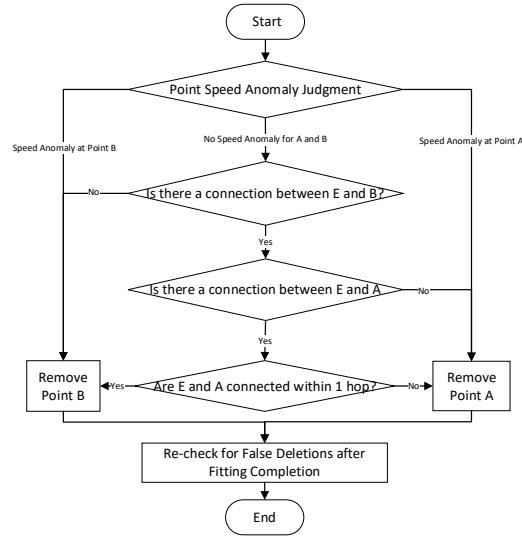


Figure 10. Head and tail noise judgment process

(2) Mid-Journey Noise Points

Targeting misidentification of hub ramp points. As shown in Figure 11, for intermediate point p_m , calculate its predecessor p_a and successor p_b , and compute hop counts $H(a,m)$, $H(m,b)$, and $H(a,b)$. Let p_a be a trusted node. If:

$$H(a,m)+H(m,b)=H(a,b) \quad (20)$$

it is considered that there is no noise point. If:

$$H(a,m)+H(m,b)>H(a,b) \quad (21)$$

then p_m is a suspected noise point and is removed. Repeat the process until all noise points are identified, then make the final determination combined with its data record status (transaction/license plate recognition) and local speed:

$$V_{local}=D_{am}/(t_m-t_a) \quad (22)$$

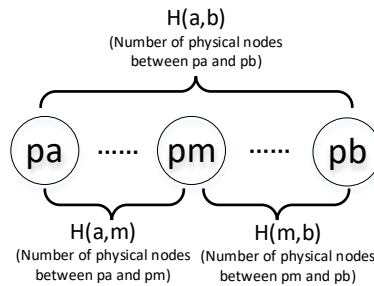


Figure 11. Schematic diagram of connectivity noise recognition

(3) Knotted Path Correction

Identify timestamp inversion caused by gantry clock errors. As shown in Figure 12, the queried gantry order is A1-B1-D1-E1-F1-C1-G1. The actual order should be

A1-B1-C1-D1-E1-F1-G1. After C1 jumps to after F1, check the front and rear gantries of C1 according to connectivity. It can be seen that C1 has two adjacent gantries, B1 and D1, so the C1 time jump is corrected.

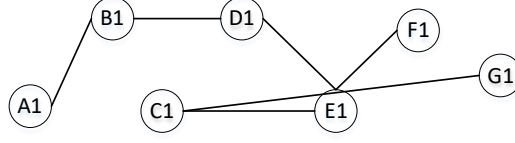


Figure 12. Schematic diagram of knotting path

(4) Overspeed Path Verification

Perform overspeed checks on all nodes in the en-route information. As shown in Figure 13, there are A, B, and C nodes in the full en-route information, with passing times t_A , t_B , t_C . The distance between A and B is $D1$, and between B and C is $D2$. The speed limit threshold is V_{limit} . Calculate the speeds between AB and BC as $V1$ and $V2$:

$$\begin{aligned} V_1 &= D_{AB} / (t_B - t_A) \\ V_2 &= D_{BC} / (t_C - t_B) \end{aligned} \quad (23)$$

Then execute the following judgments in sequence.

$$\begin{cases} \text{if } D1 \leq 10\text{Km} \wedge D2 \leq 10\text{Km} \rightarrow \text{noise-free} \\ \text{else if } V1 \leq V_{limit} \wedge V2 \leq V_{limit} \rightarrow \text{noise-free} \\ \text{else if } V1 > V_{limit} \wedge V2 > V_{limit} \rightarrow B \text{ is noise} \end{cases} \quad (24)$$

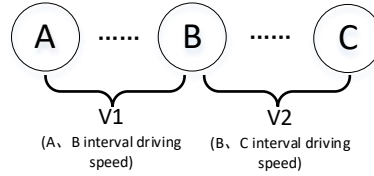


Figure 13. Schematic diagram of overspeed noise recognition

(5) Illegal U-turn Identification

Let P_{start} be the start gantry of path fitting, P_{end} the end gantry. $L(P)$: the road section number where gantry P is located. $D(P)$: direction identifier of gantry P (1 for upstream, 0 for downstream). V_{seg} : vehicle travel speed on a certain path segment.

V_{th} : normal speed threshold set by the system (implicitly ~140 km/h in this paper).

$Path_{candidate}$: set of candidate paths generated by the system. $\exists M \in Path_{candidate}$: there exists a candidate path M .

1) Discrimination Logic Formulas

Condition 1: Path abnormality check: all candidate paths have overspeed phenomena (actual travel speed exceeds the threshold):

$$\forall M \in Path_{candidate}, \max(V_{seg}(M)) > V_{th} \quad (25)$$

Condition 2: Start-end road section consistency check: the start and end points must be located on the same physical road section:

$$L(P_{start}) = L(P_{end}) \quad (26)$$

Condition 3: Travel direction opposition check: the start direction is opposite to the end direction, indicating the vehicle crossed the road centerline:

$$D(P_{start}) \neq D(P_{end}) \wedge D(P_{start}) \oplus D(P_{end}) = 1 \quad (27)$$

(Note: $\oplus$ is the XOR operation, ensuring they are strictly opposite, e.g., upstream/downstream.)

2) Comprehensive Decision Logic Formula

Only when all three conditions are satisfied simultaneously is the trajectory determined to be an illegal U-turn, triggering a specific path backfill strategy:

$$\begin{aligned} \text{IsIllegalUturn} = & \\ \begin{cases} \text{True} & \text{if } (\forall M \in Path_{candidate}, V_{\max}(M) > V_{th}) \wedge (L(P_{start}) = L(P_{end})) \wedge (D(P_{start}) \neq D(P_{end})) \\ \text{False} & \text{otherwise} \end{cases} \end{aligned} \quad (28)$$

3) Result Output (Path Fitting Rule)

If determined as an illegal U-turn ($\text{IsIllegalUturn} = \text{True}$), abandon A* optimal path search and force fitting regression according to the physically existing path in the connectivity relationship:

$$\text{FinalPath} = \text{GeneratePath}(P_{start} \rightsquigarrow P_{end} | \text{Connectivity}) \quad (29)$$

An example of illegal U-turn scenario discrimination is given below. As shown in Figure 14, A1, B1, B2, A2 are the actually queried point information. A1-B1-C1-D1-E1-F1 is the upstream direction, and A2-B2-C2-D2-E2-F2 is the downstream direction. First, according to the noise point filtering logic in the figure, judge A1-B1-B2: let M be the en-route point between A1-B1, N between B1-B2, and O between A1-B2, then $M + N = O$ ($1 + 9 = 10$). According to the overspeed path verification logic, V_1 (speed of A1-B1) is normal (80 km/h), V_2 (speed of B1-B2) is abnormal (160 km/h), and V_3 (speed of A1-B2) is abnormal (140 km/h). Similarly, it is concluded that between B1-B2-A2, the en-route noise point filtering logic is satisfied, but obvious speed anomalies also exist. Finally, judge that the start and end road section numbers are the

same: A1 and A2, B1 and B2 are two pairs of reverse gantries in opposite upstream/downstream directions, meeting the U-turn characteristics.

To sum up, when filling the path, if all candidate paths are overspeed and the start and end points are reverse gantries on the same road section, it is determined as an illegal U-turn. The overall fitting returns according to the existing path in the connectivity relationship, i.e., A1-B1-...-F1-F2-...-B2-A2.

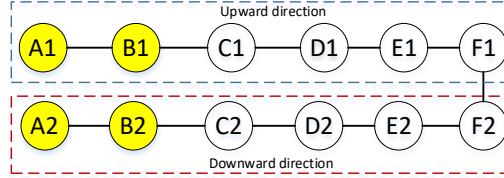


Figure 14. Diagram of Illegal U-turn

(6) Secondary Trip Identification

Identify mixed multiple trips caused by long time intervals. For consecutive points p_e, p_k , the corresponding speed is V_{ek} , distance is D_{ek} , and $T_{threshold}$ is the time difference threshold. If the time difference satisfies:

$$\Delta T = t_k - t_e > T_{threshold} \quad (30)$$

and:

$$V_{ek} = D_{ek} / \Delta T \quad (31)$$

is far lower than the reasonable level of the road network, the sequence is cut between p_e and p_k , and identified as two independent trips.

An example of secondary trip identification scenario discrimination is given below. As shown in Figure 15, the complete theoretical path is A-B-C-D-E-F-G-H-I-J-K-L-M-N. The actually received sequence is A-B-C-D-E-K-L-M-N. It is detected that the interval between E and K is 3.2 hours, the number of missing gantries is 4, and the speed is 50 km/h. It is judged that A-E is the previous trip and should be removed, returning K-N.

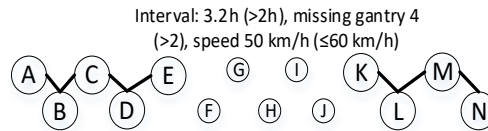


Figure 15. Secondary Stroke Schematic Diagram

2.3.2 Path Filling Based on Minimum Toll and Speed Constraints (Innovation Point)

For breakpoints still existing after deep cleaning, this paper combines the “minimum toll” business principle with physical feasibility constraints for the first time. Query the precomputed minimum toll matrix M_{COST} to obtain a candidate path list sorted in ascending order of toll amount. For each candidate path, calculate its required average speed:

$$V_{req} = D_{cand} / (t_{end} - t_{start}) \quad (31)$$

where D_{cand} is the mileage of the candidate path, t_{start} is the start time, and t_{end} is the end time. Select the first candidate path satisfying $V_{req} \leq V_{limit}$ (speed threshold). This method ensures that the lowest feasible fee is charged to the user under the premise of complying with physical laws. This step combines the minimum toll principle with the physical feasibility principle.

2.3.3 Timestamp Reconstruction

For filled nodes, linear interpolation is used to assign reasonable timestamps according to their mileage proportion between the predecessor and successor nodes. That is, the server obtains the record time of the previous node and the next node at the breakpoint, and calculates the total time difference. At the same time, obtain the total mileage of the entire fitted path from the previous node to the next node, and the cumulative mileage of each supplementary node from the previous node. Then the passing timestamp of the supplementary node is calculated as formula (32):

$$T_i = t_{start} + \frac{d_i}{D_{total}} \times (t_{end} - t_{start}) \quad (32)$$

Where d_i is the cumulative mileage of the filled node v_i from the start node v_{start} , D_{total} is the total mileage of the filled path, and t_{start} and t_{end} are the timestamps of the gantries before and after the breakpoint.

2.4 Dual-Indicator Fitting Quality Evaluation Model (Core Innovation Point)

Let the final fitted path be P_{final} . Define the following subsets: P_{final}^{orig} is the original observed gantry set; P_{final}^{unique} is the set of filled nodes with unique paths (only one feasible path); P_{final}^{ambi} is the set of filled nodes with non-unique paths.

This paper proposes a dual-indicator model for the first time to quantitatively output the reliability of path P_{final} :

(1) Confidence: Measures the certainty of path selection.

$$Conf = \frac{|P_{final}^{orig} \cup P_{final}^{unique}| \cdot W_{high} + |P_{final}^{ambi}| \cdot W_{low}}{|P_{final}| \cdot W_{high}} \times 10000 \quad (33)$$

Where W_{high} and W_{low} are weights (e.g., 10 and 5). The closer Conf is to 10000, the higher the certainty.

(2) Raw En-Route Ratio: Characterizes the dependence of the fitting result on the original data.

$$R_{raw} = \frac{|P_{final}^{orig}|}{|P_{final}|} \times 10000 \quad (34)$$

The higher this value, the fewer speculative nodes are introduced in the fitting process.

(3) Overall Calculation Reliability: We use the mean of the two indicators to represent the reliability of this path:

$$Reliability = (Conf + R_{raw}) / 2 \quad (35)$$

2.5 Experimental Verification

To verify the effectiveness of the proposed framework, the following experiments were designed.

2.5.1 Experimental Setup

(1) Dataset

More than 5 million cross-province travel records from ETC transaction data of a province in 2025 were extracted, including one month of data. About 100,000 records with missing gantries were screened out, and 1000 of them were manually reviewed and labeled as the best test set. Figure 16 shows the statistical chart of the proportion of 6 types of anomalous data.

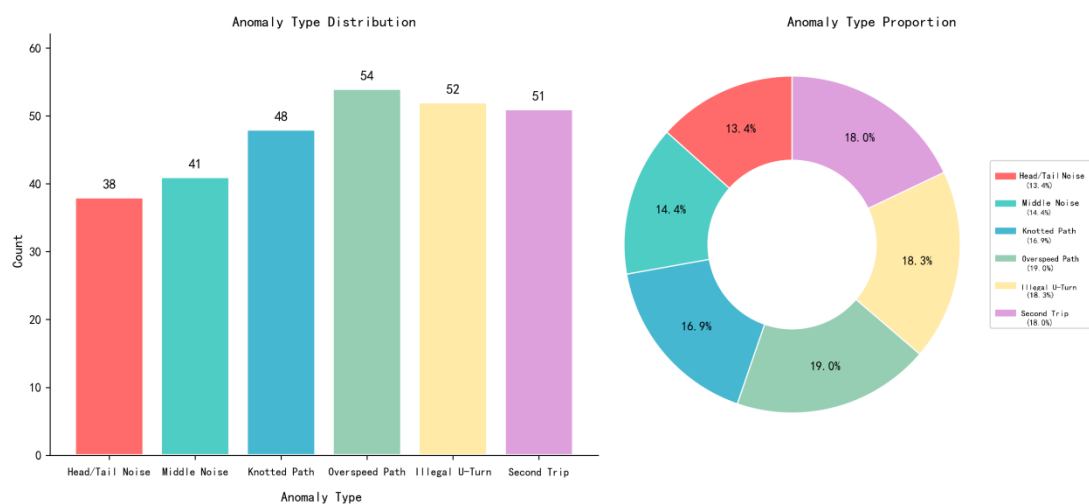


Figure 16 Statistical Chart of Anomalies of Class 16

(2) Comparison Methods

SP (Shortest Path): Traditional method, filling geometric shortest paths between breakpoints.

BL (Baseline Method): Implementation of the first stage of the framework in this paper.

Ours (Proposed Method): Complete two-stage framework.

(3) Evaluation Indicators

Fitting Accuracy: Proportion of gantry sequences where the fitted path is completely consistent with the manually labeled real path.

Billing Deviation Rate: Proportion of records where the difference between the toll calculated by the fitted path and the toll of the real path exceeds 1 yuan. This is the core indicator for measuring business fairness.

Anomaly Detection F1-score: Evaluate the recall and precision of the multi-dimensional anomaly detection engine on the best test set.

Confidence Effectiveness: Calculate the Pearson correlation coefficient between the confidence Conf and the fitting error (edit distance).

2.5.2 Result Analysis

(1) Accuracy Comparison

Figure 17 shows the accuracy comparison chart. The left chart (fitting accuracy) has three bars representing three accuracies, the higher the better. The red dashed line marks the 95% target accuracy, which only Ours exceeds. The right chart (billing deviation rate) has three bars representing the deviation rates of the three methods, the lower the better. The red arrow marks the improvement range from SP to Ours.

Table 1 shows the accuracy comparison of the three methods. The proposed method (Ours) achieves the optimal fitting accuracy. Compared with SP, it improves by 24.5 percentage points (+33.9%), and compared with BL, it improves by 12.1 percentage points (+14.3%). The deviation rate drops from 18.5% of SP to 2.1%, a decrease of 88.6%, significantly improving billing fairness. The increase in processing time is within an acceptable range.

Table 1. Accuracy Comparison

Method	Fitting Accuracy	Billing Deviation Rate	Average Processing Time (ms)	Characteristics
SP	72.3%	18.5%	15	Traditional method, only considers geometric shortest path

Method	Fitting Accuracy	Billing Deviation Rate	Average Processing Time (ms)	Characteristics
BL	84.7%	8.2%	8	First stage of this paper, guarantees connectivity
Ours	96.8%	2.1%	12	Complete two-stage framework

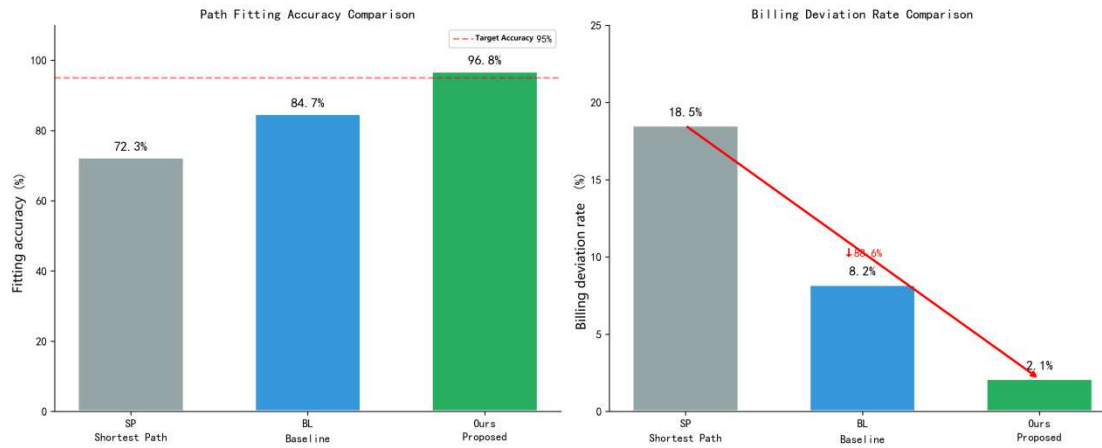


Figure 17. Accuracy Comparison Chart

(2) Confidence Distribution

Figure 18 shows the confidence analysis chart. The left scatter plot has the X-axis representing the confidence score (3000–10000), the higher the more reliable the fitting result; the Y-axis representing the path error (1–50), the lower the more accurate the fitting. The color gradient goes from green (small error) to red (large error). The trend line (red dashed line) slopes downward overall. The right chart shows that confidence is approximately normally distributed. The red dashed line (mean) and orange dot-dash line (median) almost coincide, indicating a symmetric distribution.

Key indicators are shown in Table 2. Confidence evaluation validity: the calculated Pearson correlation coefficient between confidence Conf and path fitting error is -0.87, showing a strong negative correlation. That is, paths with lower confidence have a higher probability of actual errors. When confidence > 8000, the error is usually < 10 yuan, which can be used for rapid adjudication of billing disputes. Confidence is concentrated in 6000–8500, indicating that most path fitting qualities are good. To sum up, confidence is interpretable and can provide a quantitative basis for operators.

Table 2. Key indicators of confidence

Indicator	Value	Description
Pearson Correlation Coefficient	-0.87	Strong negative correlation between confidence and error
Mean Confidence	≈ 7200	Concentrated between 6000–8500
Median Confidence	≈ 7000	Approximately normal distribution

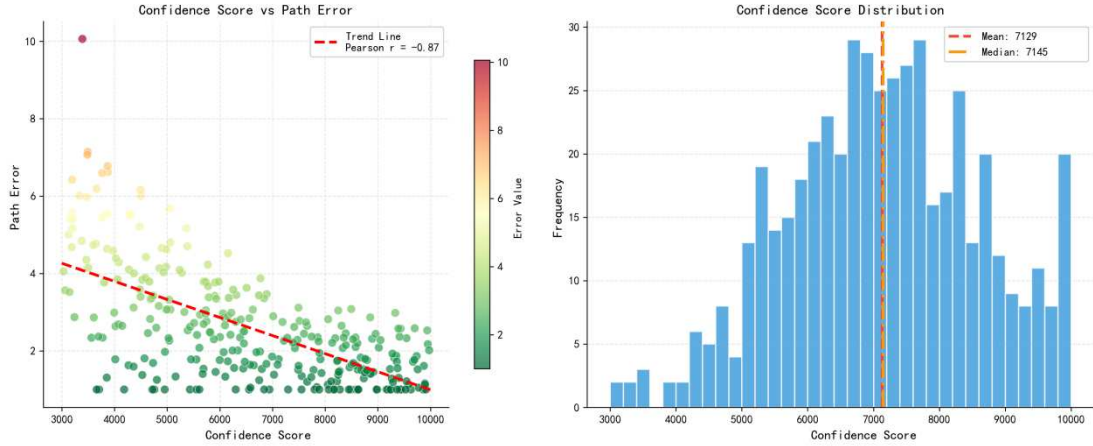


Figure 18. Confidence analysis chart

(3) Multi-dimensional Evaluation Radar Chart

Figure 19 shows the radar chart of the three methods. Five evaluation dimensions radiate from the center, with axis scales ranging from 0 to 100. Each method connects the scores of each dimension with a polyline, and the larger the area covered by the polyline, the better the overall performance. The polyline of Ours (green) expands outward obviously, covering the largest area and having the best overall performance.

Table 3 shows the five-dimensional evaluation indicators. The five-dimensional comprehensive performance of Ours is fully superior to SP and BL. SP pursues efficiency but sacrifices accuracy; Ours greatly improves accuracy (97) while maintaining high efficiency (85). Ours has a significant improvement in “interpretability” (88). The high score of Ours in “anti-interference” (95) shows that the six-type anomaly detection mechanism is effective.

Table 3. Five-dimensional Evaluation Indicators

Dimension	SP Score	BL Score	Ours Score
Accuracy	72	85	97
Stability	68	78	92
Anti-interference	65	65	95
Efficiency	90	75	85
Interpretability	50	65	88

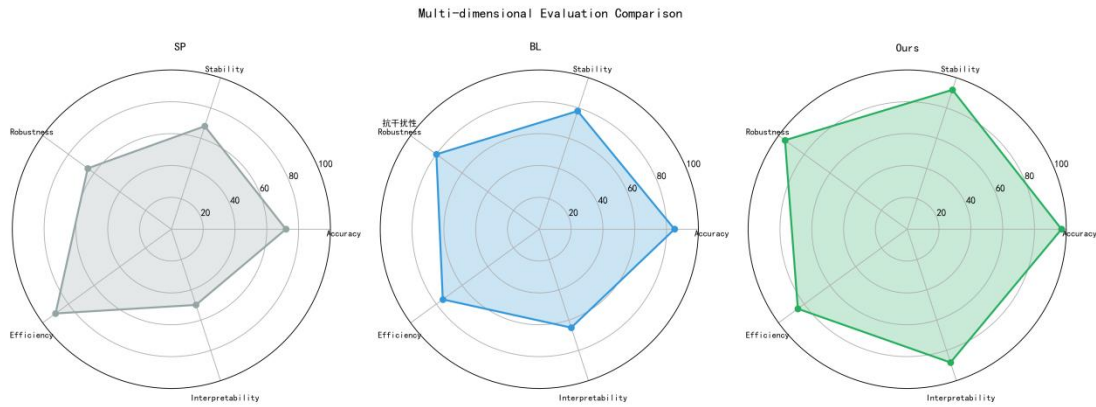


Figure 19. Comparison of Three Radar Methods

(4) Comprehensive Performance Comparison

Figure 20 shows the comprehensive performance comparison of the three methods.

Top left (processing time distribution): Three histograms stacked in red (SP), blue (BL), and green (Ours). Observe the x-axis position of the peak: BL is the leftmost (fastest), SP is the rightmost (slowest)—all three methods have processing times within an acceptable range.

Top right (anomaly detection accuracy): Blue bar (BL) vs red bar (Ours) compare the detection capabilities for various anomalies, each group has two bars, and red is higher than blue, indicating Ours is better—Ours significantly outperforms BL in detecting all 6 types of anomalies.

Bottom left (raw en-route ratio vs confidence): Red dots represent records with low raw data proportion, green dots represent those with high proportion. The upward-sloping trend line indicates a positive correlation—the more raw data, the higher the confidence.

Bottom right (cumulative error distribution): Three curves represent the cumulative distribution of different methods. The further left the curve, the smaller the deviation. Red dashed line (10 yuan) and gray dot-dash line (30 yuan) are threshold references.

Conclusions:

1) Processing efficiency: BL is the fastest (~8 ms), Ours is moderate (~12 ms), SP is the slowest (~15 ms); all are at the millisecond level, meeting real-time billing requirements.

2)Anomaly detection comprehensiveness: Ours has a detection rate $> 93\%$ for all anomaly types, significantly better than BL's 65–85%.

3)Quality evaluation reliability: Raw data proportion is positively correlated with confidence, verifying the rationality of the dual-indicator evaluation model.

4)Billing deviation control: 90% of the travel deviations of the Ours method are < 10 yuan, meeting high-precision billing requirements.

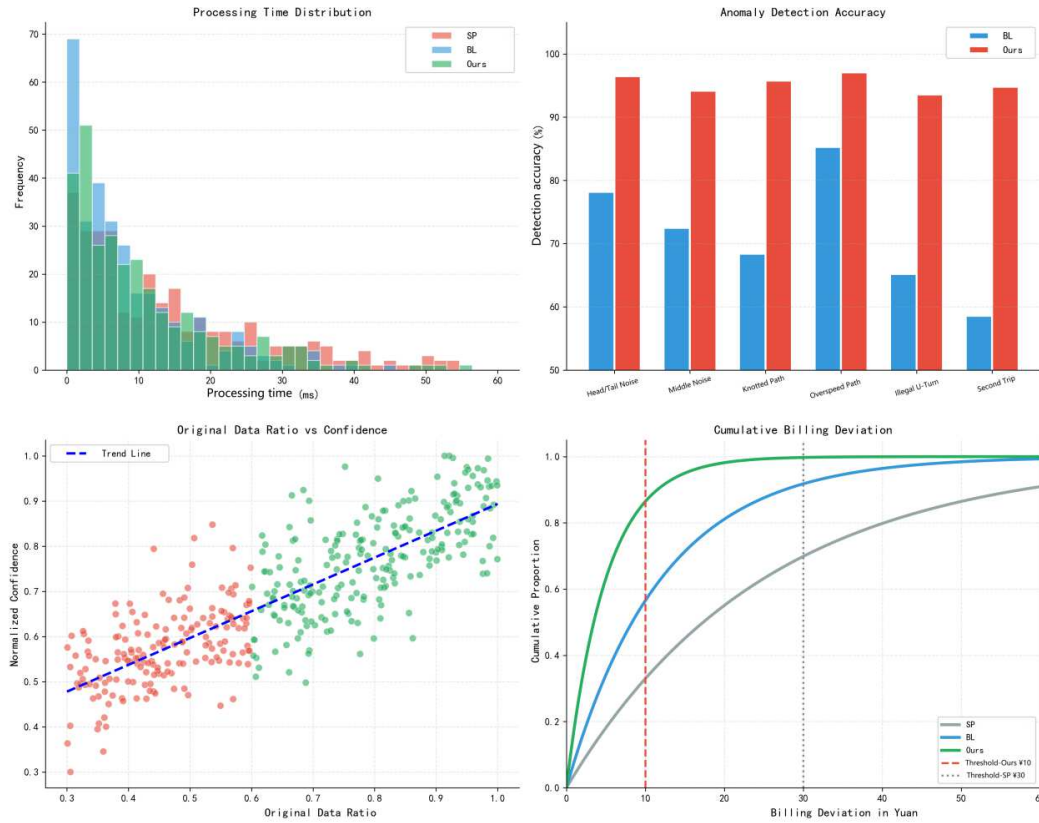


Figure 20. Comprehensive Performance Comparison of Three Methods

3 Case Studies and System Application Scenarios

3.1 Framework Application Example

The framework proposed in this paper has been integrated into the expressway network tolling system of a province. Figure 2 (in the original document) shows the visualization of the fitting process for a complex case in the system interface. The left chart is the baseline fitting (Stage I) result, where unprocessed mid-journey noise points (gray points) and wrong path branches caused by noise are still visible. The right chart is the optimized fitting (Stage II) result. After cleaning by the multi-dimensional anomaly detection engine and arbitration by the minimum toll path, wrong noise points are filtered, and the path is corrected to the correct and lowest-cost route. This instance intuitively verifies the key role of the second stage.

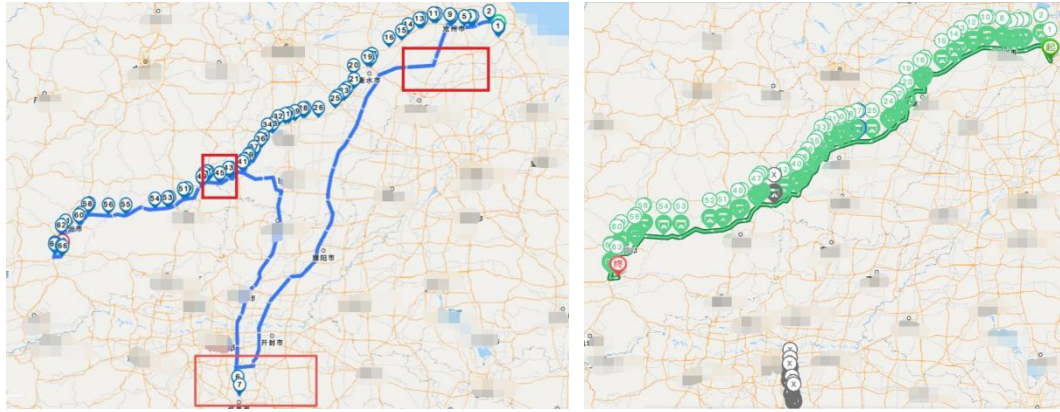


Figure 21. Example comparison of two-stage path fitting effect

3.2 Reliability Evaluation Example

As shown in Figure 21, it is the system implementation of calculating the reliability of different paths according to confidence, providing system application support for path selection during path fitting. There are 4 paths from G0610 to G0110, and 3 paths from gantry A to gantry B, among which 1 is high-reliability (reliability $\geq 94\%$), 1 is medium-reliability (90–94%), and 1 is low-reliability ($\leq 90\%$).

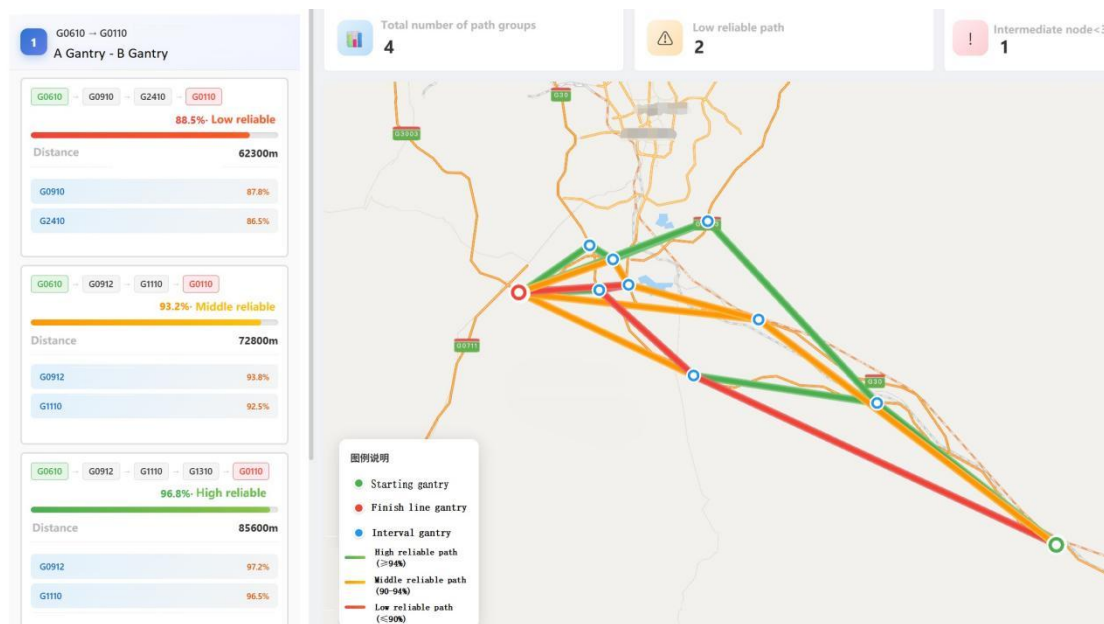


Figure 22. Reliability Calculation Example Diagram

3.3 Future Work

Current work mainly targets passenger cars (Class I vehicles). Future work will focus on: (1) Extending the framework to truck billing by incorporating special rules such as vehicle type rate coefficients and bridge/tunnel tolls into the calculation of the minimum toll matrix; (2) Using massive historical trajectory big data to build a more refined path selection probability model to further optimize arbitration among

ambiguous paths; (3) Exploring the integration of real-time traffic status information (e.g., congestion) to dynamically adjust path screening based on speed constraints.

4 Conclusion

Aiming at the demand for accurate expressway billing, this paper proposes an innovative two-stage path fitting optimization framework. The main contributions and conclusions of this framework are as follows:

(1) A two-stage methodology of “connectivity guarantee → business rule optimization” is proposed, systematically embedding the minimum toll principle into the path fitting process for the first time, solving the problem of disconnection between technical optimization and business objectives.

(2) A parallelized multi-dimensional anomaly detection engine is designed, completing mathematical modeling and collaborative detection for six typical expressway anomalies for the first time, significantly improving fitting robustness under complex data environments.

(3) A dual-indicator fitting quality evaluation model is established, providing quantifiable reliability indicators for path fitting results for the first time, enhancing system interpretability and credibility.

(4) Experiments based on real data show that this framework not only improves the path fitting accuracy to 96.8%, but also reduces the billing deviation rate by nearly 70%, greatly guaranteeing billing fairness while improving accuracy. The strong correlation (-0.87) between the confidence indicator and error verifies the effectiveness of the evaluation model.

This framework provides a solid theoretical cornerstone and practical solution for building accurate, fair, and transparent expressway network tolling systems, and has important application and promotion value.

5 Funding Statement

This study was supported by the National Natural Science Foundation of China (Grant number 62373367).

Acknowledgments

We would like to express our sincere gratitude to Mr. Li Bin, Director of the Road Network Monitoring and Emergency Response Center, for his continuous support and guidance. We appreciate the excellent research environment and abundant resources he has generously provided, which served as the fertile ground for the successful completion of this work.

Data Availability Statement

The datasets generated and/or analysed during the current study are not publicly available due to [their nature as proprietary operational data] but are available from the corresponding author on reasonable request.

References

- [1] Ministry of Transport of the People's Republic of China. Operation and Service Rules for Expressway Network Tolling. 2019.
- [2] Wang Yunpeng, et al. Intelligent Vehicle-Road Cooperative Systems. Science Press, 2019.
- [3] Zheng Y. Trajectory data mining: an overview. *ACM Transactions on Intelligent Systems and Technology*, 2015, 6(3): 1-41.
- [4] Chen Xiao, Yang Xiaoguang, Li Keping. Research on travel time reliability of urban expressways based on ETC data. *Journal of Transportation Systems Engineering and Information Technology*, 2018, 18(2): 40-46.
- [5] Quddus M A, Ochieng W Y, Noland R B. Current map-matching algorithms for transport applications: State-of-the-art and future research directions. *Transportation Research Part C: Emerging Technologies*, 2007, 15(5): 312-328.
- [6] Goh C Y, Dauwels J, Mitrovic N, et al. Online map-matching based on hidden Markov model for real-time traffic sensing applications. In: *IEEE International Conference on Intelligent Transportation Systems*. 2012.
- [7] Lou Y, Zhang C, Zheng Y, et al. Map-matching for low-sampling-rate GPS trajectories. In: *ACM SIGSPATIAL GIS*. 2009.
- [8] Dijkstra E W. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1959, 1(1): 269-271.
- [9] Hart P E, Nilsson N J, Raphael B. A formal basis for the heuristic determination of minimum cost paths. *IEEE Transactions on Systems Science and Cybernetics*, 1968, 4(2): 100-107.
- [10] Hunter T, Abbeel P, Bayen A. The path inference filter: model-based low-latency map matching of probe vehicle data. *IEEE Transactions on Intelligent Transportation Systems*, 2013, 15(2): 507-519.
- [11] Wang H, Tang J. A Bayesian network approach for vehicle path prediction at intersections using historical trajectory data. *IET Intelligent Transport Systems*, 2019, 13(5): 846-853.
- [12] AutoNavi, et al. White Paper on Smart Expressway Path Restoration and Fee Audit Technology Based on Multi-source Data. 2022.
- [13] Hu Zhihua, Xie Binglei, Gao Ziyu. Expressway network travel time estimation considering data quality. *China Journal of Highway and Transport*, 2020, 33(8): 205-214.
- [14] Li Jianguo, Wang Wei, Sun Jian. Review of automatic detection methods for urban road traffic abnormal events. *China Journal of Highway and Transport*, 2019, 32(4): 17-30.
- [15] Liu Xiaoming, et al. Key Technologies of Vehicle-Road Cooperative Systems. People's Communications Publishing House, 2018.
- [16] Yuan J, Zheng Y, Zhang C, et al. An interactive-voting based map matching algorithm. *MDM*, 2010.
- [17] Jin Sheng, et al. Review of incomplete trajectory data repair methods. *Journal of Software*, 2019, 30(2): 363-381.

- [18] Liu Z, Zhang B, Zhang M, et al. A quantitative method for the analysis of ship collision risk using AIS data. *Ocean Engineering*, 2023, 272: 113906.
- [19] Wang Lei, Zhao Xiangmo, Liu Zhanwen. Research progress on traffic data quality assessment and control in intelligent connected environment. *Journal of Traffic and Transportation Engineering*, 2021, 21(1): 1-20.
- [20] He Zhaocheng, et al. *Urban Traffic Control Intelligence*. People's Communications Publishing House, 2020.
- [21] Yang Dongyuan, Liu Wangbao. *Traffic Big Data*. People's Communications Publishing House, 2017.
- [22] Lu Huapu, et al. *Urban Traffic Big Data*. Science Press, 2016.
- [23] Traffic Management Research Institute of the Ministry of Public Security. *National Trunk Highway Network Traffic Operation Monitoring and Evaluation Report*. 2020.
- [24] Shao Chunfu, et al. *Principles of Traffic Planning*. China Railway Publishing House, 2018.
- [25] Pei Yulong, et al. *Traffic Flow Theory*. People's Communications Publishing House, 2017.
- [26] Sun Lijun, et al. *Analysis of Road Capacity*. People's Communications Publishing House, 2015.
- [27] Guo Xiucheng, et al. *Traffic Engineering*. Southeast University Press, 2011.
- [28] Li L, et al. A survey of path planning and trajectory optimization for autonomous vehicles. *IEEE Transactions on Intelligent Vehicles*, 2022.
- [29] Zhang J, Wang F Y. Parallel driving in CPSS: A unified approach for transport automation and intelligence. *IEEE/CAA Journal of Automatica Sinica*, 2021.
- [30] China Intelligent Transportation Association. *China Smart Highway Development Report*. 2023.
- [31] Chen C, et al. Digital twin for transportation systems: A survey. *IEEE Access*, 2021.
- [32] Wang Jianqiang, et al. *Traffic State Perception and Prediction in Internet of Vehicles Environment*. Tsinghua University Press, 2022.
- [33] Liu B, Soares C G. Recent developments in ship collision analysis and challenges to an accidental limit state design method. *Ocean Engineering*, 2023, 270: 113636.
- [34] Zhang Wei, Liu Jianping, Wang Yun. Short-term traffic flow prediction model based on attention mechanism LSTM. *Journal of Transportation Systems Engineering and Information Technology*, 2021, 21(1): 60-66.
- [35] Kim S J, Taimuri G, Kujala P, et al. Comparison of numerical approaches for structural response analysis of passenger ships in collisions and groundings. *Mar. Struct.*, 2022, 81: 103125.