Optimizing Emergency Response: Intelligent Routing Decision Support System for First Responders at Rail Crossings

Yuche Chen, PhD
Principal Investigator
Department of Civil and Environmental Engineering
University of South Carolina

Yu Qian, PhD
Co-Principal Investigator
Department of Civil and Environmental Engineering
University of South Carolina

Ning Ding
Graduate Research Assistant
Department of Civil and Environmental Engineering
University of South Carolina

Dimitrios C. Rizos, PhD
Associate Director, UTCRS, Co-Principal Investigator
Department of Civil and Environmental Engineering
University of South Carolina

A Report on Research Sponsored by

University Transportation Center for Railway Safety (UTCRS)

Molinaroli College of Engineering and Computing
University of South Carolina

September 2025

Technical Report Documentation Page

1. Report No. UTCRS-USC-O5CY24	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Optimizing Emergency Response: Intelligent Routing Decision Support System for First Responders at Rail Crossings		5. Report Date September 30, 2025 6. Performing Organization Code UTCRS-USC
7. Author(s) Yuche Chen, Yu Qian, Ning Ding, and Dimitrios Rizos		8. Performing Organization Report No. UTCRS-USC-O5CY24
9. Performing Organization Name and Address University Transportation Center for Railway Safety (UTCRS) University of South Carolina (USC) Columbia, SC 29208		10. Work Unit No. (TRAIS) 11. Contract or Grant No. 69A3552348340
12. Sponsoring Agency Name and Address U.S. Department of Transportation (USDOT) University Transportation Centers Program 1200 New Jersey Ave. SE Washington, DC, 20590		13. Type of Report and Period Covered Project Report June 1, 2024 – August 31, 2025 14. Sponsoring Agency Code USDOT UTC Program
15. Supplementary Notes		

16. Abstract

The expansion of urban railway networks improves mobility but simultaneously increasing unpredictability in traffic systems particularly due to significant delays experienced by first responders at level crossings blocked by passing trains. While existing routing systems may account for train-induced blockage durations, they often overlook the additional delays caused by post-blockage traffic dissipation, thereby limiting their reliability in time-critical emergency scenarios. This study aims to provide a more accurate estimation of total delay time at level crossings by incorporating both train blockage time and traffic decongestion time into routing decision systems to improve the overall effectiveness and reliability of emergency response operations.

17. Key Words Routing; Railroad Operations; Emergency Response Time	18. Distribution Statement This report is available for download from https://www.utrgv.edu/railwaysafety/research/operations/index.htm		
19. Security Classification (of this report) None	20. Security Classification (of this page) None	21. No. of Pages 34	22. Price

Table of Contents

List of Figures

List of Abbreviations

GKDE	Gaussian Kernel Density Estimation
GMM	Gaussian Mixture Model
VAE	Variational Autoencoder

Disclaimer

Acknowledgements

The authors wish to acknowledge the University Transportation Center for Railway Safety (UTCRS) for funding this project under the USDOT UTC Program Grant No. 69A3552348340.

1. SUMMARY

The urban infrastructure of Columbia, SC features a dense and interwoven network of railways and roadways, leading to a high concentration of at-grade railroad crossings. Train movements at these intersections frequently cause significant congestion, placing considerable strain on the city's transportation system, which is particularly critical for first responders, potentially undermining emergency response effectiveness.

To address this challenge, this study proposes an intelligent routing decision support system, which is designed to dynamically reconfigure emergency vehicle routes when the planned path is obstructed by the train at grade crossing. By proactively identifying optimal alternatives, the system will minimize delays caused by rail-induced traffic disruptions and enhance the overall efficiency of emergency response operations.

The total delay estimation module employs a Monte Carlo-based joint sampling approach to predict the probability density distribution of delay time based on the historical GPS data. The estimated delay is then averaged and forwarded to the rerouting module, which determines whether rerouting is necessary and provides an optimized alternative path.

The evaluation results verify that the proposed system is both robust and practically effective. Through diverse simulation scenarios, we validate the seamless integration between delay estimation and dynamic path planning. Compared to traditional emergency routing strategies, our method significantly reduces response time, improves route efficiency, and alleviates traffic congestion. In particular, the system achieves up to 79.27% reduction in response time compared to wait-and-stop strategies, confirming its operational benefits. These advantages highlight the strong potential of our system for deployment in intelligent transportation systems and urban railway safety management.

2. BACKGROUND

The expansion of urban railway systems has led to the proliferation of grade crossings in cities. While essential for connectivity, these crossings frequently disrupt traffic flow, causing severe delays during train passages. For first responders, such disruptions can critically hinder emergency operations, putting lives and property at risk[1]. These challenges underscore the necessity of

intelligent routing systems that can adaptively guide first responders through grade crossings amid dynamic and uncertain traffic conditions.

The current route decision systems are typically composed of the delay time prediction module and the decision planning module. The former estimates potential delays caused by train crossings, while the latter uses this information to determine the optimal route for first responders.

Accurately and comprehensively predicting the total traffic delay experienced by first responders due to train blockages at grade crossings is fundamental to the effectiveness of the overall decision planning system. Existing research on train delay prediction can be broadly categorized into three methodological approaches: analytical methods, simulation-based methods, and data-driven methods[2].

Analytical methods construct simplified mathematical models, using explicit formulas and logical reasoning to approximately estimate system behaviors such as train delays or blockage times under specific assumptions. Frank[3] proposed a model where trains are assumed to travel at constant speeds, with travel times following a uniform distribution, and estimated blockage times at grade crossings based on the accumulation of trains when demand exceeds capacity. While this method provides a preliminary framework for simplified scenarios, it overlooks dynamic speed variations and the potentially skewed distribution of actual travel times. Peterson[4] extended the model by incorporating overtaking, speed variability, and priority rules to improve delay estimation accuracy. Higgins et al.[5] proposed a model to quantify the expected positive delay for individual passenger trains and track segments in urban rail networks. Although these improvements enhanced adaptability to basic scheduling rules, they still rely on assumptions of operational regularity. Consequently, analytical models struggle to capture the complex, dynamic congestion phenomena inherent in modern railway systems.

Simulation-based methods involve constructing virtual system models to dynamically replicate real-world train operations within computational environments, thereby enabling the evaluation of train delay characteristics. Peterson et al.[6] and Dessouky et al.[7] developed fine-grained simulation approaches that model railway infrastructure and dynamic operational rules to simulate delay propagation across various scenarios, including single-track and double-track lines, junctions, and terminals. These methods effectively capture both the dynamic behaviors of

individual trains and the complex interactions among multiple trains. However, simulation models inherently rely on extensive computational experiments to approximate delay time distributions. As a result, these systems often entail high computational costs, slow processing speeds, and significant modeling complexity, while remaining limited to predefined scenarios and lacking the flexibility to generate diverse blockage time distributions.

Data-driven methods utilize large volumes of historical observational data to extract underlying patterns through model construction, enabling the prediction of train delays or blockage times. These methods can be broadly classified into parametric and non-parametric models[8]. Parametric models assume predefined functional forms and focus on estimating model parameters; examples include linear regression[9], Bayesian networks[10], and time series models[11]. Although structurally simple and computationally efficient, parametric models struggle to capture complex nonlinear relationships among variables. To overcome this limitation, non-parametric models, which adapt model structures based on data without assuming fixed functional forms, have been introduced. Representative approaches such as K-Nearest Neighbors (KNN)[12], Random Forests, and Recurrent Neural Networks (RNNs) and their variants like Long Short-Term Memory (LSTM) networks[13] have demonstrated strong capabilities in modeling dynamic and nonlinear patterns. Nevertheless, traditional data-driven methods remain heavily reliant on historical data and often fail to generalize to unseen scenarios, due to their limited ability to generate new data, resulting in reduced adaptability under emerging or extreme conditions.

Moreover, most existing intelligent rerouting systems estimate delay time only based on train blockage duration at grade crossings. This oversimplified assumption overlooks other critical dynamic factors, often resulting in substantial discrepancies between estimated and actual total delay times, thereby undermining the accuracy of route selection and the effectiveness of emergency response strategies.

After delay times are predicted and output, the decision planning module dynamically generates optimal routing strategies based on the real-time state of the transportation network, aiming to guide first responders efficiently to emergency sites. To this end, transportation networks are typically modeled as graphs comprising nodes and edges, with graph search algorithms widely employed for path planning[14]. This modeling framework has become a fundamental component in intelligent transportation systems.

Graph-based path planning methods have been extensively developed, encompassing several classical algorithms. Dijkstra's algorithm[15], one of the earliest and most effective shortest path search methods, iteratively expands the node with the smallest tentative distance, guaranteeing the optimal path in graphs with non-negative edge weights. The A* algorithm[16] builds upon Dijkstra's approach by introducing heuristic evaluation functions to estimate the cost to the destination, thereby accelerating the search process in goal-oriented scenarios. Meanwhile, the Bellman-Ford algorithm[17] addresses graphs with arbitrary edge weights through iterative relaxation of edges; however, its computational efficiency is considerably lower compared to Dijkstra and its variants, particularly in large-scale networks.

In transportation systems, where road segment weights are typically non-negative and networks are large and highly dynamic, Dijkstra's algorithm has become one of the most widely adopted methods due to its high computational efficiency and robust stability. It provides a solid theoretical and practical foundation for fast and reliable path generation in urban transportation, navigation systems, and emergency response operations.

Nevertheless, in dynamic environments where train blockages and traffic congestion cause frequent changes in edge weights, the traditional Dijkstra algorithm has limited ability to adapt quickly to evolving traffic conditions. The label correcting method[18], an extension and variant of Dijkstra's algorithm, addresses this issue by allowing node labels to be updated multiple times during the search process, thereby enabling more flexible path adjustments under dynamic conditions. This method not only preserves Dijkstra's reliability for shortest path computation in non-negative graphs but also enhances computational efficiency and adaptability through optimized relaxation strategies, making it particularly suitable for large-scale dynamic transportation networks.

Overall, the development of an intelligent path decision system specifically designed for first responders navigating grade crossing scenarios holds significant practical value. At the same time, substantial opportunities remain for improving the design and performance of each core module within such systems, which constitutes the primary focus of this research.

3. OBJECTIVES

This work is dedicated to developing a novel intelligent routing system designed to support first responders in efficiently navigating scenarios involving railway grade crossings. The primary objective is to accurately predict the total delay time caused by train-induced blockages and to optimize emergency travel routes, thereby reducing delays in critical response operations. The specific contributions of this work are as follows:

1. A novel joint sampling framework is developed based on the M/M/1 traffic flow model and implemented using the Monte Carlo method to estimate the total delay time experienced by first responders in railway blockage scenarios. This framework integrates three key time distributions: train blockage time, modeled using a VAE-GMM approach; and vehicle arrival time and queue clearance time, both modeled with Gamma distributions. By jointly sampling from and synthesizing these distributions, the system constructs a probability density distribution of total delay time. Sampling and averaging from this distribution further enables accurate and robust delay time estimation
2. Based on the estimated total delay time, this study employs the Label Correcting algorithm to optimize the travel paths of first responders when encountering blocked grade crossings to effectively reduce response delays in emergency operations.

4. METHODOLOGY

4.1 Train GPS Data Analysis

The historical data set used in this study was provided by CSX Transportation and includes the latitude and longitude of railroad grade crossings in the Columbia metropolitan area of South Carolina, along with GPS trajectory data from 37 trains operating in the region between July 13 and July 20, 2020. The GPS data contain detailed information such as timestamps, train and locomotive identifiers (Train ID and Loco ID), train lengths, speeds, geographic coordinates, and heading directions of individual cars. The grade crossing coordinates were first imported into ArcGIS Pro for spatial visualization, enabling an analysis of train trajectory patterns across the crossings, as illustrated in Figure 1.

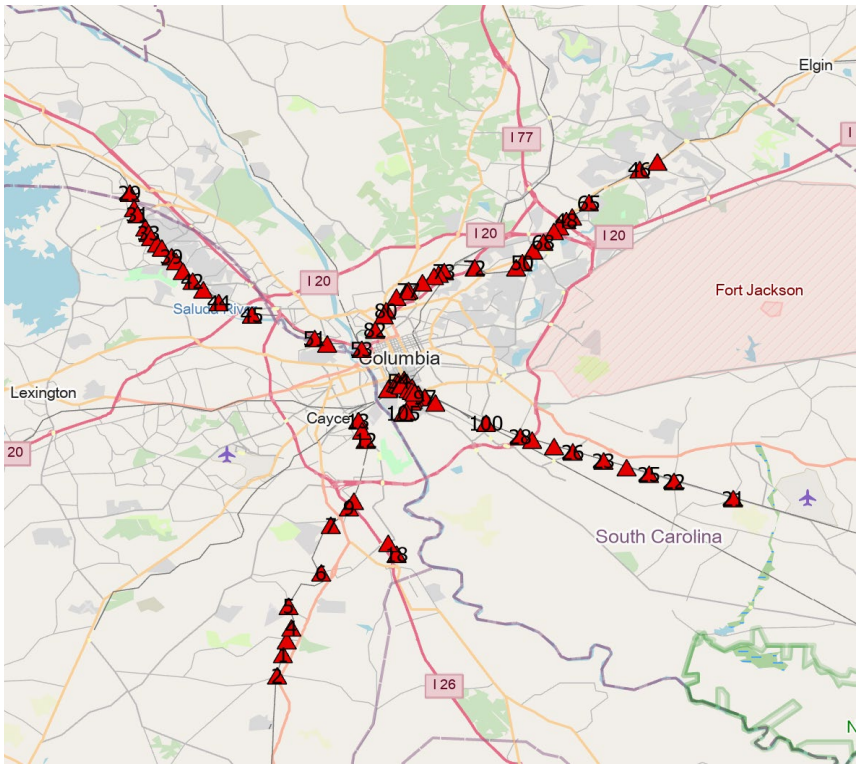
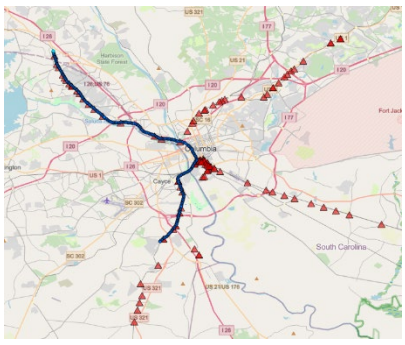
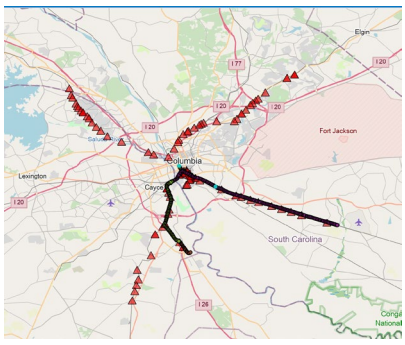


Figure 1 Grade Crossings Data in the CSV Dataset

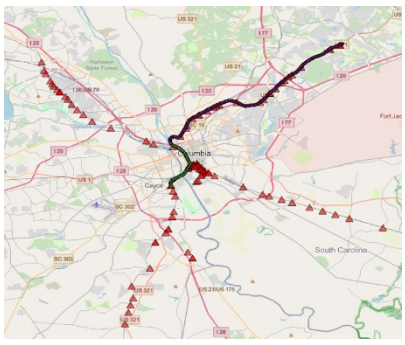
Subsequently, the longitude and latitude coordinates of train trajectories from the dataset were imported into ArcGIS Pro for processing. Based on 29,158 GPS records, the primary distribution patterns of train operations were visualized, as shown in Figure 2. The analysis results indicate that the train trajectories can be categorized into six typical patterns, with the proportions of each pattern being: (a) 15.3%, (b) 10.9%, (c) 5.6%, (d) 7.9%, (e) 34.2%, and (f) 18.7%, respectively. (Note: For security reasons, specific train identifiers and their corresponding travel routes are not disclosed in this study.)



Northwest–South



Southeast



Southwest–Northeast

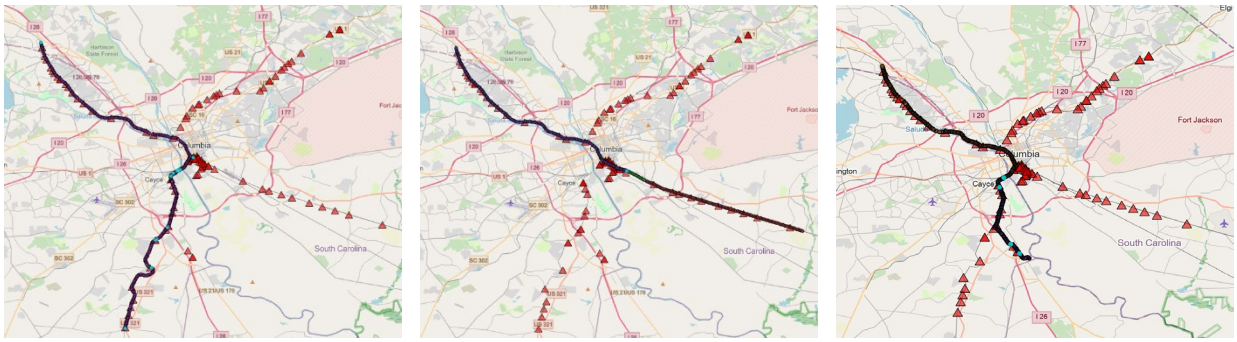


Figure 2 Train Trajectories in the CSX Dataset

The original dataset includes train length information for 37 different trains. Figure 3 presents a scatter plot of these lengths, with red points highlighting the maximum length observed for each train.

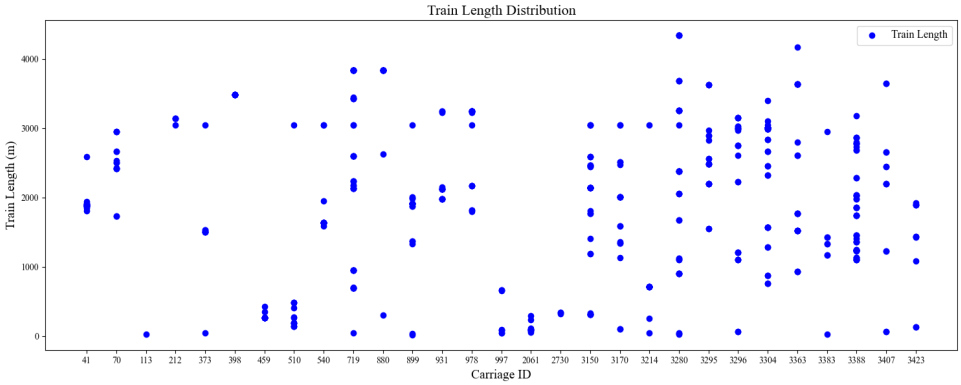


Figure 3 Scatter Plot of Train Length

Table 1 summarizes the statistical characteristics of the train lengths, which range from 40 feet to 17,028 feet. The average train length is 5,719 feet, with a standard deviation of 3,773 feet.

Table 1 Descriptive Statistics of Train Length

Index	Value(m)
Mean	1743.22
Standard Deviation	1150.06
Min	12.19
25 % Quantiles	914.40
Median	1660.55
75 % Quantiles	2749.91
Max	5190.13

Figures 4 and 5 respectively present the scatter plot and histogram of instantaneous speeds based on the 29,158 GPS records, illustrating the variations across different time points and train cars.

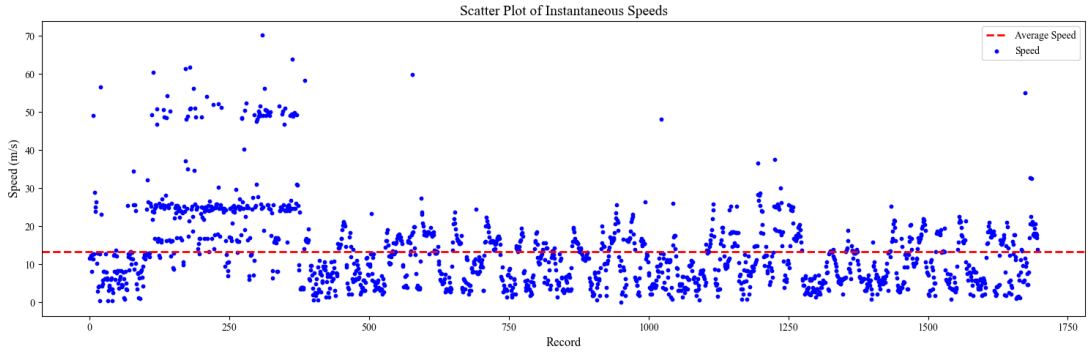


Figure 4 Scatter Plot of Instantaneous Speed

In the scatter plot of instantaneous train speeds, the red line indicates the overall average speed across all recorded data points, calculated to be approximately 13.83 meters per second.

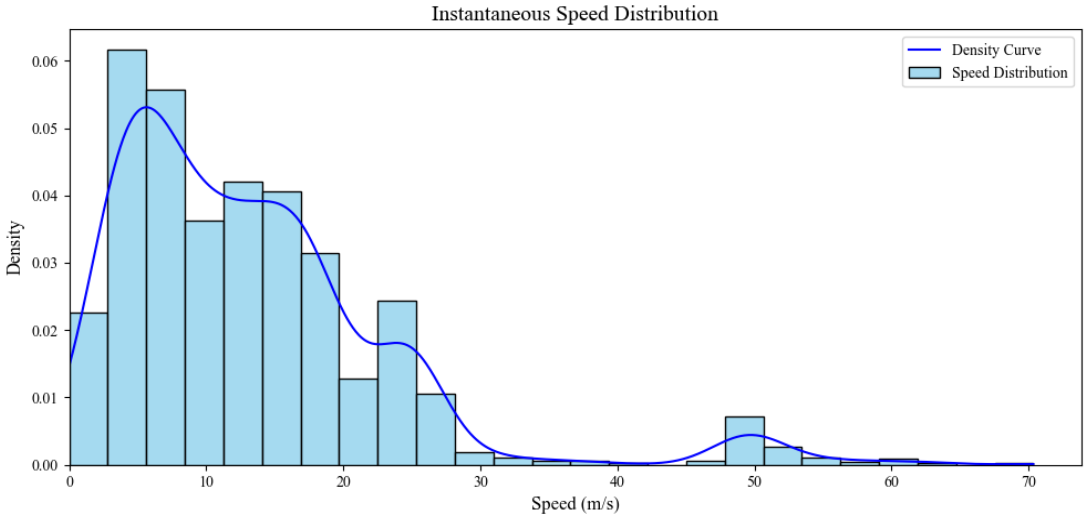


Figure 5 Histogram of Train Instantaneous Speed Distribution

As shown in Figure 5, instantaneous train speeds range from 0 to 73 meters per second. The distribution exhibits a right-skewed pattern, with most speeds concentrated between 0 and 30 meters per second, and a pronounced peak around 5 meters per second. Several secondary peaks appear between 10 and 20 meters per second, suggesting that trains generally operate at lower speeds within the Columbia metropolitan area. Additionally, a smaller cluster of speeds is observed between 45 and 55 meters per second, potentially corresponding to specific trains maintaining higher velocities along certain urban segments.

4.2 Blockage Time Estimation

To accurately estimate blockage windows at railway crossings, the initial step is identifying the critical GPS data point on the train trajectory near the crossing location. To achieve this, each train's trajectory data is first imported into ArcGIS software for spatial visualization, enabling precise determination of the relative positions between trajectory points and the railway crossing.

Specifically, given the railway crossing coordinates as $C(x_C, y_C)$ a spatial search region is defined as a circle with an initial radius a. The mathematical formulation of this search region Ω is expressed as:

$$\Omega = \left\{ (x, y) \mid \sqrt{(x - x_c)^2 + (y - y_c)^2} \le a \right\} \tag{1}$$

Within the defined region Ω, it is necessary to verify the positional relationship between the GPS trajectory data points and the railway crossing, specifically determining whether each data point lies along the train's movement direction towards the crossing. To accomplish this, pairs of consecutive trajectory points, denoted as $P_i(x_i, y_i)$ and $P_{i+1}(x_{i+1}, y_{i+1})$, are analyzed sequentially. From these points, we define the train's traveling direction vector $\vec{V}_i$:

$$\vec{V}_i = (x_{i+1} - x_i, y_{i+1} - y_i) \tag{2}$$

Simultaneously, the position vector from trajectory point P_i to the crossing location $C(x_C, y_C)$ is formulated as vector $\vec{U}_i$:

$$\vec{U}_i = (x_C - x_i, y_C - y_i) \tag{3}$$

The positional relationship of each trajectory point relative to the crossing is then assessed by calculating the dot product of these two vectors. The decision criterion is defined as:

If $\vec{V}_i \cdot \vec{U}_i > 0$, then trajectory point P_i is approaching the railway crossing along the travel direction.

Among all the trajectory points within region Ω satisfying the above criterion, the closest point to the railway crossing is selected as the critical GPS point and labeled as P_A.

If no suitable trajectory point can be found within the initial search radius r, the radius is dynamically adjusted by incrementally enlarging its size until at least one eligible trajectory data point is located. This dynamic radius adjustment mechanism can be mathematically defined as follows:

$$r_{k+1} = r_k + \Delta r, \;\; k = 0,\ 1,\ 2,\ \ldots \tag{4}$$

Here, r_0 denotes the initial radius, and the incremental value Δr is typically set at 10 m. The above iterative process continues until a suitable GPS data point is found. Once the critical GPS data point P_A is identified, further blockage time estimation procedures at railway crossings can be subsequently conducted based on this established reference point.

After determining the critical GPS point P_A on the train trajectory approaching the railway crossing, the subsequent step is to identify the GPS point closest to the train's position when it has fully traversed the crossing. To achieve this, we initiate the calculation from the railway crossing $C\left(x_C, y_C\right)$, with the initial reference point set at P_A, the closest GPS trajectory point approaching the crossing.

The initial distance between the railway crossing C and the trajectory point P_A, denoted as d_0 is calculated using the Haversine formula:

$$d_0 = 2R\,arcsin\left(\sqrt{sin^2\left(\frac{\varphi_A - \varphi_C}{2}\right)} + cos\left(\varphi_A\right)cos\left(\varphi_C\right)sin^2\left(\frac{\lambda_A - \lambda_C}{2}\right)\right) \tag{5}$$

Where R is earth's radius and (φ_A, λ_A) and (φ_C, λ_C) represent latitude and longitude of points P_A and C, respectively, converted into radians.

Then, starting from point P_A and along the train's traveling direction, we continuously accumulate distances between consecutive GPS data points:

$$D_{cum} = d_0 + \sum_{j=A}^{n-1} d_{(j,j+1)} \tag{6}$$

Here, the distance between consecutive points P_j and P_{j+1} is similarly calculated using the Haversine formula.

The accumulation continues until the cumulative distance D_{cum} approximates the train's total length l_{train}:

$$D_{cum} \approx l_{train} \tag{7}$$

Upon satisfying this condition, we record the GPS data point P_B and its timestamp t_B. The timestamp of the initial critical GPS point P_A is denoted as t_A.

Next, we calculate the time interval Δt between timestamps t_A and t_B:

$$\Delta t = t_B - t_A \tag{8}$$

Then, based on cumulative distance D_{cum} and time interval Δt, we determine the average speed v_{avg} during the train's crossing:

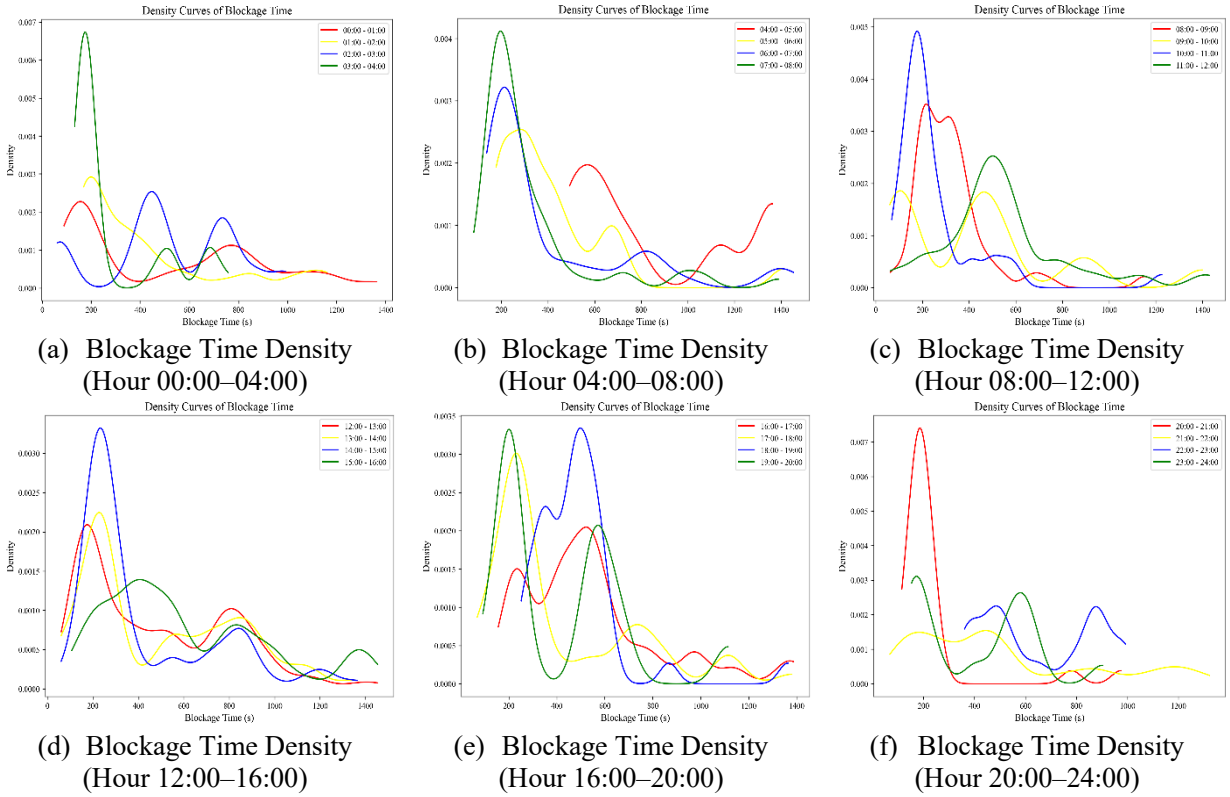
$$v_{avg} = \frac{D_{cum}}{\Delta t} \tag{9}$$

Finally, using train length l_{train} and calculated average speed v_{avg}, the precise blockage time $T_{blockage}$ for the railway crossing can be estimated as:

$$T_{blockage} = \frac{l_{train}}{v_{avg}} \tag{10}$$

4.3 Blockage Time Result

Based on the GPS trajectory data and the proposed methodology, a total of 1,125 valid blockage time instances were identified. The 24-hour day was divided into hourly intervals, and blockage times were statistically analyzed within each interval. To capture the temporal variations in blockage durations, Gaussian Kernel Density Estimation (KDE) was applied to smoothly fit the blockage time distributions for each hour. The resulting density curves are shown in figure 6.



(a) Blockage Time Density (Hour 00:00–04:00)

(b) Blockage Time Density (Hour 04:00–08:00)

(c) Blockage Time Density (Hour 08:00–12:00)

(d) Blockage Time Density (Hour 12:00–16:00)

(e) Blockage Time Density (Hour 16:00–20:00)

(f) Blockage Time Density (Hour 20:00–24:00)

Figure 6 Blockage Time Density Distribution

4.4 Joint Sampling Module

4.4.1 Blockage Time Distribution

Blockage times at railway crossings often exhibit complex, multimodal distributions. However, due to practical limitations in data collection, historical blockage time data frequently contain sparsely populated or missing regions. Traditional single-distribution models, such as the normal distribution, are inadequate for capturing such intricacies. While non-parametric methods like Kernel Density Estimation (KDE) can fit observed data directly, they are susceptible to substantial bias in data-sparse regions, limiting their generalizability.

To address these challenges, this study adopts a generative modeling approach that combines Variational Autoencoders (VAE) with Gaussian Mixture Models (GMM), referred to as VAE-GMM. This framework is designed to capture multimodal characteristics and mitigate sampling bias inherent in sparse datasets. The VAE is a probabilistic generative model rooted in machine learning that teaches a latent variable distribution to effectively encode and reconstruct input data. Specifically, the encoder maps input data **x** to a continuous latent space to obtain representations

z, which are then decoded back into the original space. The objective of the VAE is to maximize the Evidence Lower Bound (ELBO), defined as follows:

$$\mathcal{L}_{VAE}(\phi, \theta) = \mathbb{E}_{q_\phi(\mathbf{z} \mid \boldsymbol{x})}[log_{p_\theta}(\boldsymbol{x} \mid z)] - \mathrm{KL}\ (q_\phi(z \mid \boldsymbol{x}) \parallel p(\mathbf{z})) \tag{11}$$

where ϕ and θ represent parameters of the encoder and decoder networks, respectively, $q_\phi(\boldsymbol{z} \mid \boldsymbol{x})$ is the approximate posterior distribution, $p_\theta(\boldsymbol{x} \mid z)$ is the likelihood function defined by the decoder, and KL denotes the Kullback–Leibler divergence. Optimizing this function enables VAE to learn the underlying distribution and generate new samples.

The Gaussian Mixture Model (GMM) is a powerful probabilistic approach capable of precisely modeling multimodal distributions through a linear combination of multiple weighted Gaussian distributions:

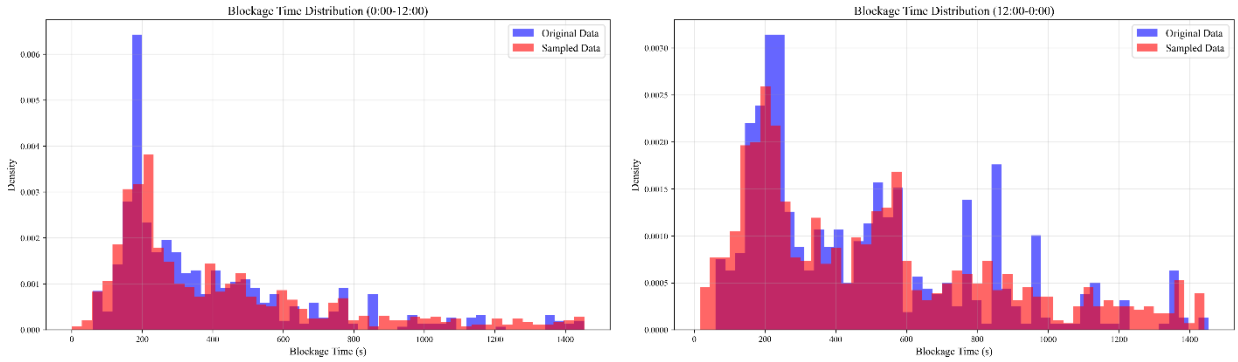
$$p(\boldsymbol{x}) = \sum_{k=1}^{K} \pi_k \boldsymbol{\mathcal{N}}\ (\boldsymbol{x} \mid \mu_k, \sum k), \text{ where } \sum_{k=1}^{K} \pi_k = 1 \tag{12}$$

In this formulation, K denotes the number of Gaussian components, and π_k, μ_k, and $\sum k$ represent the weights, means, and covariance matrices of the *k-th* component, respectively.

By integrating GMM with VAE, specifically setting GMM as a prior distribution for latent variables allows VAE to effectively capture complex nonlinear data structures and explicitly model multiple peaks in blockage time distributions.

4.4.2 VAE-GMM Sampling Result

In this experiment, the blockage time data computed in Section 4.2 were used as the input for training the VAE-GMM model.



(a) Sampled Data Distribution (0:00-12:00)

(b) Sampled Data Distribution (12:00-0:00)

Figure 7 VAE-GMM Sampled Data Distribution

The dataset was split into 70% for training and 30% for validation. For experimental design, the 24-hour day was divided into two time periods: 00:00–12:00 and 12:00–24:00. The blockage time data within each period were modeled and analyzed separately.

Figures 7 (a) and (b) illustrate the original blockage time distributions for two different time periods. In each figure, the blue histogram represents the distribution of the original dataset, while the red histogram shows the corresponding samples generated by the VAE-GMM model. As shown in the figures, the original data exhibits typical multimodal structures and long-tail characteristics. In particular, the early hours contain dense clusters of blockage events, while the later periods feature sparse and more oscillatory patterns.

The VAE-GMM model successfully captures the peak density characteristics in both time periods and demonstrates strong generalization in the tail regions. More importantly, the model not only preserves the statistical properties of the original data but also generates statistically reasonable new samples that do not exist in the original dataset. This enables the model to effectively fill in gaps in the data space, particularly in low-frequency regions.

4.4.3 Arrival Time Distribution

In most existing studies on delay estimation, researchers tend to consider only the delay caused by train blockages at railway crossings, assuming that first responders are always affected for the full duration of the blockage. However, this approach is often incomplete and may lead to overestimation of the actual impact on emergency response times. In real-world scenarios, a queue of vehicles typically forms upstream of the crossing once it is blocked by a train. This means that if a first responder arrives at the crossing sufficiently after the blockage has begun, they may not need to wait for the entire blockage duration. Consequently, this introduces variability into the rerouting process.

To address this limitation, the present study incorporates vehicle arrival time into the estimation of total delay time, enabling a more comprehensive and accurate modeling of delays experienced by first responders.

In the field of traffic flow modeling, the M/M/1 queuing model is widely used to characterize the statistical properties of vehicle arrival and queuing behavior. The first "M" denotes that vehicle arrivals follow a Markovian Poisson process, meaning the number of vehicle arrivals in any given time interval follows a Poisson distribution, and the inter-arrival times follow an exponential

distribution. The second "M" indicates that the service time, such as the time required for a vehicle to pass through a control point like a toll booth or a grade crossing, also follows an exponential distribution. The "1" signifies that there is a single service channel in the system.

This model effectively captures the sequential arrival pattern of vehicles ahead of a first responder when a railway crossing is blocked by a passing train.

$$f_T(t) = \lambda e^{-\lambda t}, \ t \geq 0 \tag{13}$$

In the equation, λ represents the average vehicle arrival rate, and its reciprocal $1 / \lambda$ denotes the mean inter-arrival time.

Given that the inter-arrival times between consecutive vehicles follow an exponential distribution, our interest lies in the total time required for the arrival of the *n-th* vehicle. This total time, denoted as $S_n = \sum_{i=1}^{n} T_i$, represents the sum of independent exponentially distributed intervals starting from the first vehicle. According to properties in probability theory, the sum S_n follows a Gamma distribution with shape parameter n and rate parameter λ. Its probability density function is given by:

$$f_{S_n}(t) = \frac{\lambda^n t^{n-1} e^{-\lambda t}}{(n-1)!}, \ t \geq 0 \tag{14}$$

Based on this probability density function, we can model the arrival time of any individual vehicle within the waiting queue when a train blocks the railway crossing.

4.4.4 Arrival Time Distribution Results

When constructing the Gamma distribution to model cumulative vehicle arrival times, one of the key parameters is the average vehicle arrival rate λ. In this study, a data-driven approach is adopted to estimate this parameter. Specifically, field data collected at the Catawba Street railway crossing are used to analyze the relationship between blockage duration and the corresponding queue length. The number of vehicle arrivals associated with various blockage durations in the dataset is illustrated in figure 8.

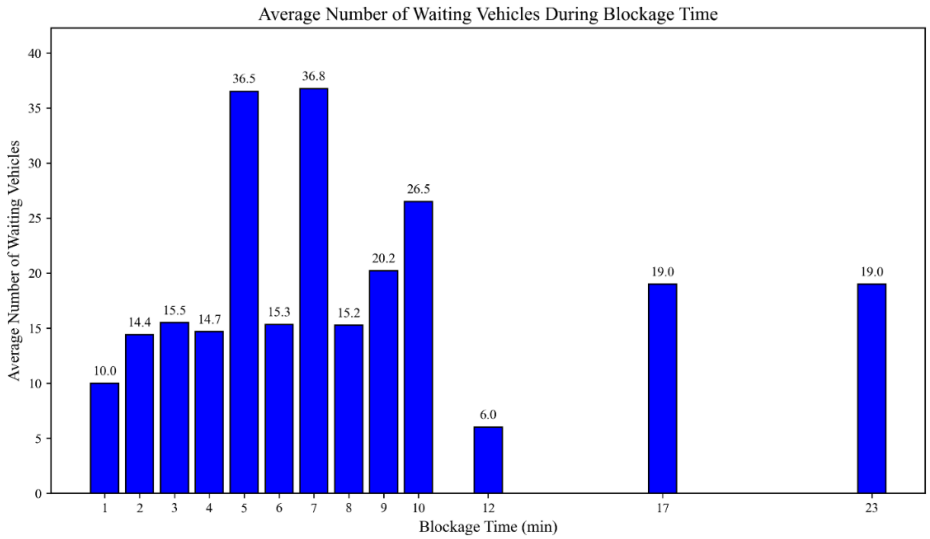
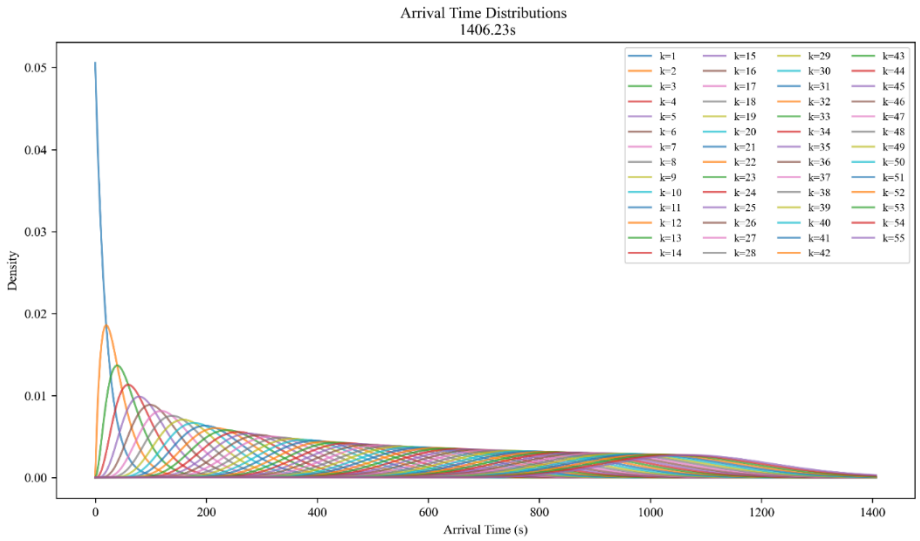


Figure 8 Average Number of Waiting of Vehicles During Blockage Time

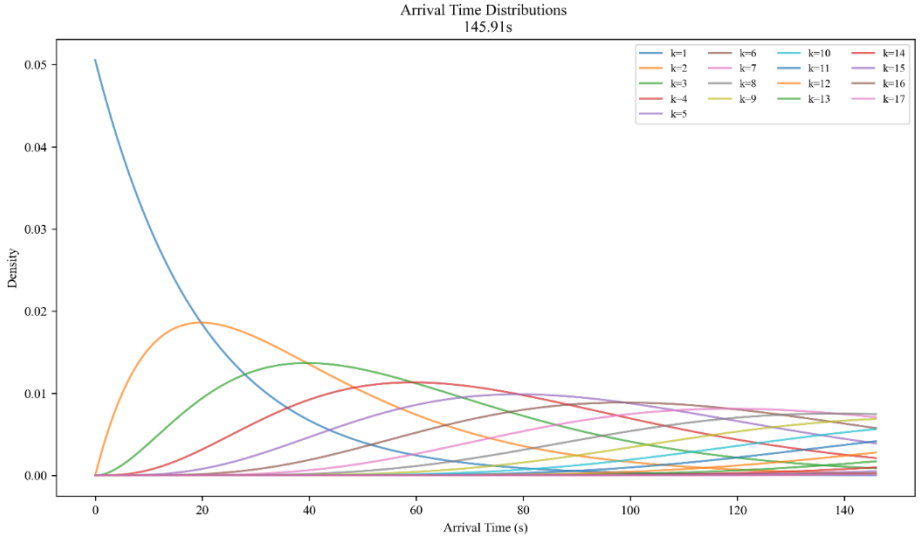
In this study, we focus on the global average arrival rate λ, which represents the average number of vehicles arriving per unit time across all blockage events. Specifically, let N_i denote the number of vehicles observed during the *i-th* blockage event, and T_i the corresponding blockage duration. The global average arrival rate can then be defined as:

$$\lambda = \frac{\sum_{i=1}^{n} N_i}{\sum_{i=1}^{n} T_i} \tag{15}$$

By computing the global average arrival rate, we avoid the local fluctuations introduced by modeling each blockage event individually, allowing for a more consistent representation of the overall vehicle arrival intensity within the system. Once the global average arrival rate λ is obtained, it is incorporated into the Gamma distribution to model the probability density of arrival times for different vehicles within a blockage queue. To demonstrate the applicability and expressiveness of this modeling approach, we plotted the Gamma probability density curves corresponding to different vehicle counts under varying blockage durations.



(a) Arrival Time Gamma Distributions (Blockage Time = 1406.23 s, k= 1∼55)



(b) Arrival Time Gamma Distributions (Blockage Time = 145.91 s, k = 1∼17)

Figure 9 Arrival Time Distributions

Figure 9 (a) and (b) illustrate the cumulative vehicle arrival time distributions constructed using Gamma distributions under two different blockage durations. Each curve represents the arrival time distribution of the *k-th* vehicle within the queue during the blockage period. The maximum number of vehicles that can be modeled depends on the product of the blockage duration and the average arrival rate.

For longer blockage durations, for example, 1,406.23 seconds in the figure 4.9(a), the system can model arrival distributions for up to k = 55 vehicles. In contrast, for shorter blockage durations such as 145.91 seconds in figure 4.9(b), the model can only capture the arrival patterns for the first

$k = 17$ vehicles. Vehicles beyond this threshold are assumed to arrive after the blockage window has ended and, therefore, are not subject to delay caused by the train.

4.4.5 Clearance Time Distribution

In railway grade crossing scenarios, the passage of a train marks the end of the physical obstruction, but not the full recovery of the traffic system. The queue of vehicles formed during the blockage requires additional time to dissipate after the crossing reopens, as traffic flow gradually returns to normal. This recovery process, beginning with the movement of the first vehicle at the front of the queue and ending with the last vehicle clearing the crossing, is referred to as clearance time.

Omitting clearance time from total delay modeling leads to systematic underestimation of actual delays, which can misguide routing decisions and result in suboptimal path choices for first responders.

In the M/M/1 queuing model, clearance time corresponds to the service phase. During the train's occupation of the crossing, traffic flow is interrupted, and vehicles continue to arrive, forming a queue. Once the crossing reopens, vehicles begin to pass through the bottleneck at a certain rate, initiating the service phase as the queue gradually dissipates.

In this phase, it is assumed that the time intervals between the departures of consecutive vehicles follow an exponential distribution. By summing these exponential intervals, a Gamma distribution is obtained, representing the probability density of the clearance time for the first n vehicles. In this context, the rate parameter λ of the Gamma distribution corresponds to the service rate, indicating the number of vehicles passing through the crossing per unit time.

4.4.6 Clearance Time Distribution Results

To compute the global average service rate λ, we rely on field data collected at the Catawba Street railway crossing. This dataset not only records the number of vehicles queued during each blockage event, but also captures the time required for all queued vehicles to clear the crossing from the moment the crossing gate is raised to the passage of the last vehicle.

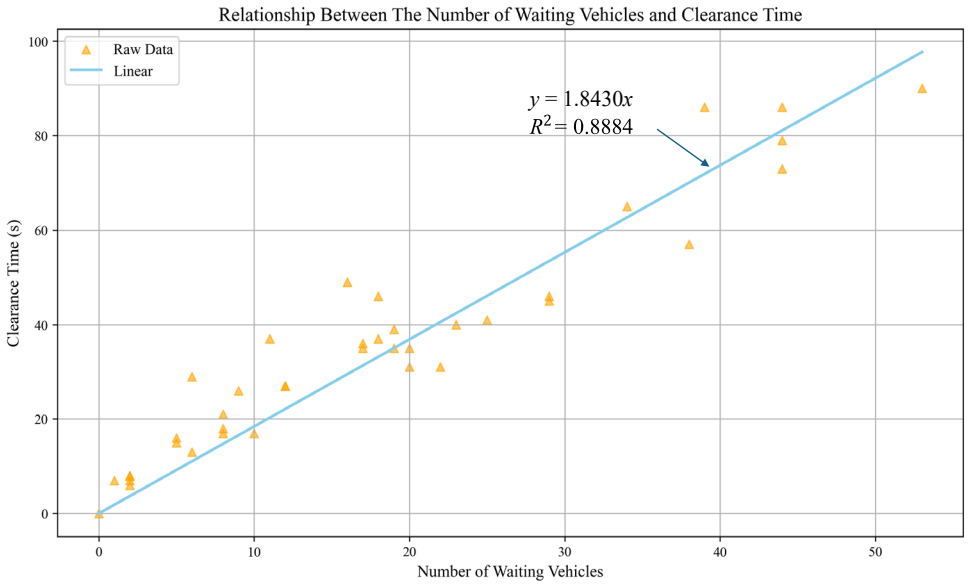


Figure 10 Relationship Between the Number of Waiting Vehicles and Clearance Time

In figure 10, the orange triangle markers in the figure represent the observed discrete relationship between the number of queued vehicles and the corresponding clearance time during historical blockage events. To quantify this relationship, a linear regression model constrained to pass through the origin was applied to all observed data points. The resulting best-fit line is $y = 1.8430x$, where x denotes the number of vehicles waiting and y the corresponding clearance time. The coefficient of determination $R^2 = 0.8884$ indicates that the linear model effectively captures the overall trend between vehicle count and clearance time.

The slope parameter $a = 1.843$ can be interpreted as the average clearance time per vehicle. Its reciprocal yields the service rate, $\lambda \approx 0.5426$ vehicles per second. By incorporating this service rate into the Gamma distribution, we derive the probability density curves of clearance time for individual vehicles at different queue positions k, as shown in figure 11.

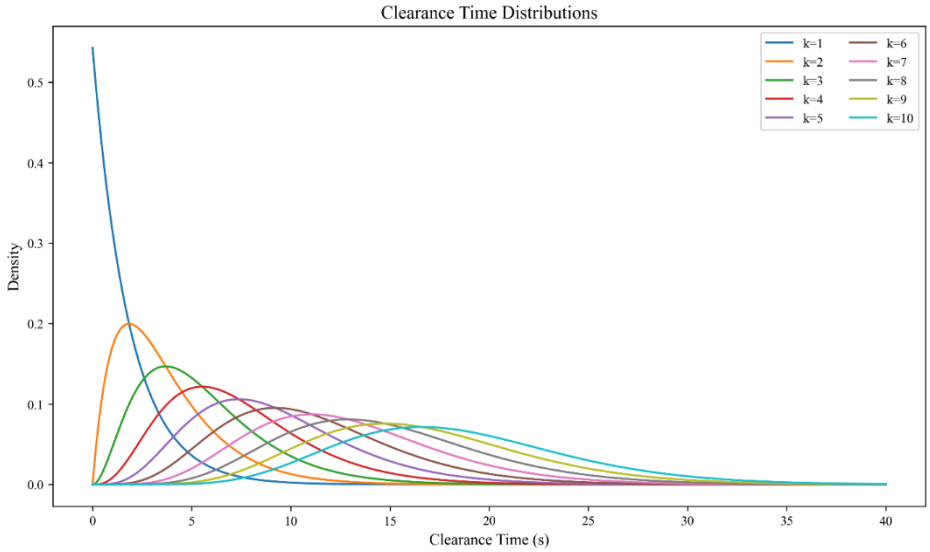


Figure 11 Clearance Time Gamma Distributions (k = 1∼10)

To illustrate the modeling performance, the figure presents the clearance time probability density curves for a queue of 10 vehicles, covering vehicles from position 1 to 10. It is important to note, however, that this modeling approach is highly scalable. Regardless of the actual queue length, the corresponding clearance time distributions can be generated using the global service rate λ, enabling fine-grained temporal estimates for congestion dissipation under varying traffic conditions.

4.4.7 Joint Sampling Model Results

Based on the distributions described above, we first sample blockage times at railway crossings using the trained Variational Autoencoder–Gaussian Mixture Model (VAE-GMM), conditioned on the train's traversal time. To ensure the validity of the sampled data, only those samples falling within the 95% confidence interval of the learned blockage time distribution are retained.

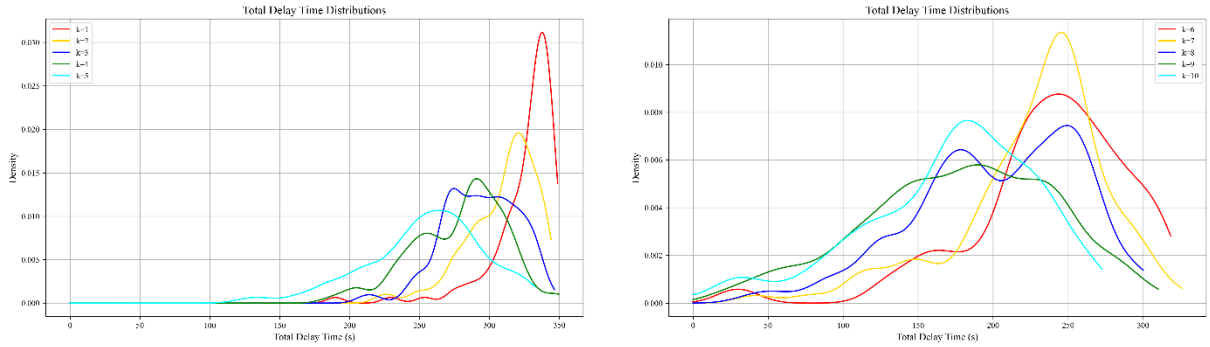
For each sampled blockage time, Gamma distribution models are then constructed to represent the arrival time and clearance time of vehicles, depending on their position within the queue. Using Monte Carlo simulation, sets of samples are drawn from these distributions. Based on these samples, the total delay time is computed using the following formulation:

$$Total\ Delay\ Time = Blockage\ Time - Arrival\ Time + Clearance\ Time \tag{16}$$

A large set of total delay time samples is generated and fitted using Gaussian Kernel Density Estimation (GKDE) to obtain a smooth probability distribution. To mitigate the impact of

individual sample variability, samples are drawn from the fitted curve and average, improving the robustness and reliability of the delay estimation.

In the first case, we assume that the train arrives at the railway crossing between 12:00 a.m. and 12:00 p.m. The sampled blockage time in this case is 343.2495 seconds. We then set the number of Monte Carlo samples for both the arrival time and clearance time distributions to 1000. The first responder is assumed to be positioned within the queue at k = 1 to 10. The resulting total delay time distribution is shown as follows:

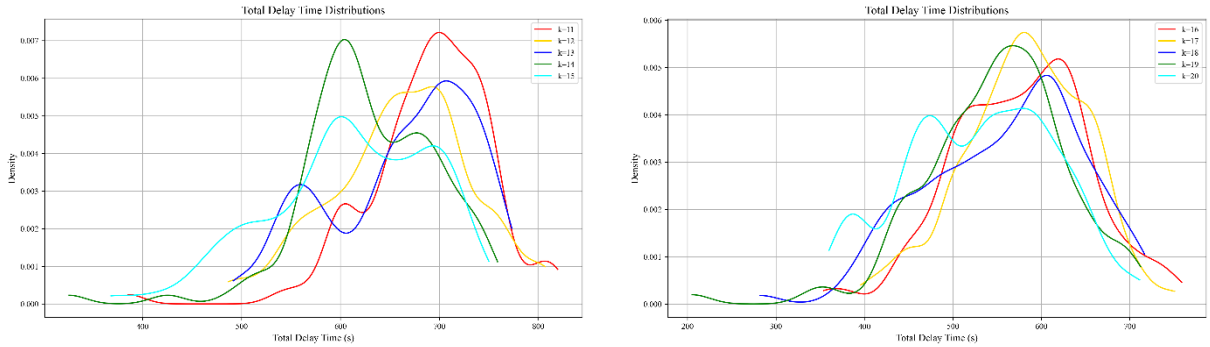


(a) Total Delay Time Distributions (k = 1~5) (b) Total Delay Time Distributions (k = 6~10)

Figure 12 Total Delay Time Distributions

As shown in Figure 12(a), vehicles positioned at the front of the queue exhibit peak total delay times concentrated between 280 and 320 seconds, with k = 5 showing the most pronounced peak. This indicates that these vehicles were stopped early during the blockage, resulting in longer and more concentrated delays. In contrast, vehicles in positions k = 6 to 10 experienced significantly shorter delays, with peaks generally between 180 and 270 seconds and flatter distributions. Overall, the closer a vehicle arrives to the end of the blockage period, the less delay it tends to experience.

In the second case, we assume that the train arrives at the railway crossing between 1:00 p.m. and midnight. The sampled blockage time in this case is 878.8104 seconds. As in the previous scenario, we draw 1000 samples from the arrival time and clearance time distributions. The first responder is assumed to be positioned within the queue at k = 11 to 20. The resulting total delay time distribution is shown as follows:



(a) Total Delay Time Distributions (k = 11~15) (b) Total Delay Time Distributions (k = 16~20)

Figure 13 Total Delay Time Distributions

Figure 13(a) and (b) illustrate the total delay time distributions for vehicles positioned at k = 11 to 20 when the blockage duration extends to 878.81 seconds. Compared to the previous case, where the blockage lasted 343.25 seconds, the distribution peaks have noticeably shifted to the right, mostly falling between 540 and 700 seconds. This shift reflects the overall increase in waiting time caused by the longer blockage duration.

4.5 Route Planning for First Responder at Grade Crossing

In emergency response systems, the path planning algorithm plays a central role. By receiving the total delay time estimated from the joint sampling module, it identifies the shortest path within the road network from the responder's current location to the emergency site, aiming to minimize response time and enhance rescue efficiency.

The presence of railway blockages introduces inherently unpredictable delays, posing significant challenges for conventional shortest path algorithms that assume static edge weights. These traditional methods often lack the adaptability required to respond to dynamic traffic disruptions, thereby limiting their effectiveness in time-sensitive emergency response scenarios. In contrast, the Label Correcting algorithm, grounded in dynamic programming theory, exhibits strong flexibility in accommodating time-varying and context-dependent edge costs. As such, it is particularly well-suited for routing problems under uncertainty, such as those induced by stochastic train-induced obstructions at grade crossings.

We begin by considering a directed graph $G = (V, A)$, where V denotes the set of nodes and $A \subseteq V \times V$ represents the set of directed edges. For any edge $(i, j) \in A$ (i, j), if the corresponding road segment intersects a railway grade crossing, the joint sampling model is employed to estimate the

total delay time experienced by vehicles traversing that segment. This process yields a set of delay samples $\{d_{ij}^{(1)}, d_{ij}^{(2)}, d_{ij}^{(3)}, \ldots, d_{ij}^{(n)}\}$.

To mitigate the influence of outliers and stochastic fluctuations in single samples, the mean of these delay samples $\hat{D}_{ij}$ is computed and used as the predicted delay cost for edge (i, j). This averaging process improves the stability and reliability of path cost estimation under uncertainty.

Accordingly, the total travel cost of a path segment can be redefined as:

$$C_{ij} = t_{ij} + \hat{D}_{ij} \tag{17}$$

t_{ij} is the baseline travel time from node i to node j; $\hat{D}_{ij}$ is the estimated railway-induced delay on edge (i, j), obtained from the output of the joint sampling model.

The algorithm begins by initializing the path cost label $d(v)$ for all nodes $v \in V$ to infinity, except for the source node r, which is assigned a label of zero. A candidate node queue Q is then initialized, containing only the source node.

During the path search process, the algorithm repeatedly extracts the current node u from the queue and examines all its adjacent edges (u, v). If the cost of reaching node v via node u, computed as $d(u)+ C_{uv}$, is less than the currently recorded label $d(v)$, then $d(v)$ is updated accordingly. Node v is then added to the queue if it is not already present.

This iterative process continues until no further updates occur and the queue becomes empty. Through this label updating mechanism, the algorithm progressively converges to the shortest path solution while allowing for flexible adjustments to path costs in environments with dynamically changing edge weights.

To further formalize the path search as an optimization problem, we introduce a binary decision variable $x_{ij}^{rs} \in \{0, 1\}$, which indicates whether edge (i, j) is included in the path from source node r to destination node s. The objective is to minimize the total path cost, defined as:

$$\min_{x} \sum_{(i,j)\in A} C_{ij} \cdot x_{ij}^{rs} = \min_{x} \sum_{(i,j)\in A} \left(t_{ij} + \hat{D}_{ij}\right) \cdot x_{ij}^{rs} \tag{18}$$

where C_{ij} represents the composite travel cost on edge (i, j), incorporating both baseline travel time and estimated railway-induced delay.

To ensure both path connectivity and uniqueness, the following flow balance constraints must be satisfied:

$$\sum_{j:(i,j)\in A} x_{ij}^{rs} - \sum_{j:(i,j)\in A} x_{ji}^{rs} = \begin{cases} 1, & if\ i = r \\ -1, & if\ i = s \\ 0, & otherwise \end{cases} \tag{19}$$

This constraint guarantees that there is a single directed path from the source node r to the destination node s, with continuous flow through intermediate nodes.

Additionally, the decision variable is defined as:

$$x_{ij}^{rs} = \begin{cases} 1, & if\ edge\ (i,j)\ lies\ on\ the\ path\ from\ r\ to\ s \\ 0, & otherwise \end{cases} \tag{20}$$

In summary, by integrating the enhanced Label Correcting path planning module with the total delay time estimation module, an optimized routing system is established. This system enables first responders to identify the most efficient path to their destination while accounting for delays caused by train-induced blockages at railway grade crossings.

4.6 Route Optimization at Grade Crossing Results

In this section, we present the results of our intelligent route optimization system for emergency response in downtown Columbia, South Carolina. Specifically, we examine how the system dynamically replans optimal paths when railway level crossings along the initially planned route are blocked by passing trains, comparing scenarios where emergency calls occur in the morning versus the afternoon.

For instance, consider an emergency call occurring at 10:02 AM, during which a train has already departed from the southeastern area and is making a turn through the triangular railway network in downtown Columbia, heading southwest. In this scenario, the emergency occurs near a residential building on Catawba Street, while the nearest available police unit is located on Heyward Street. Our system first generates an initial shortest-time path for the responder to reach the caller. The planned route is illustrated in the figure below.

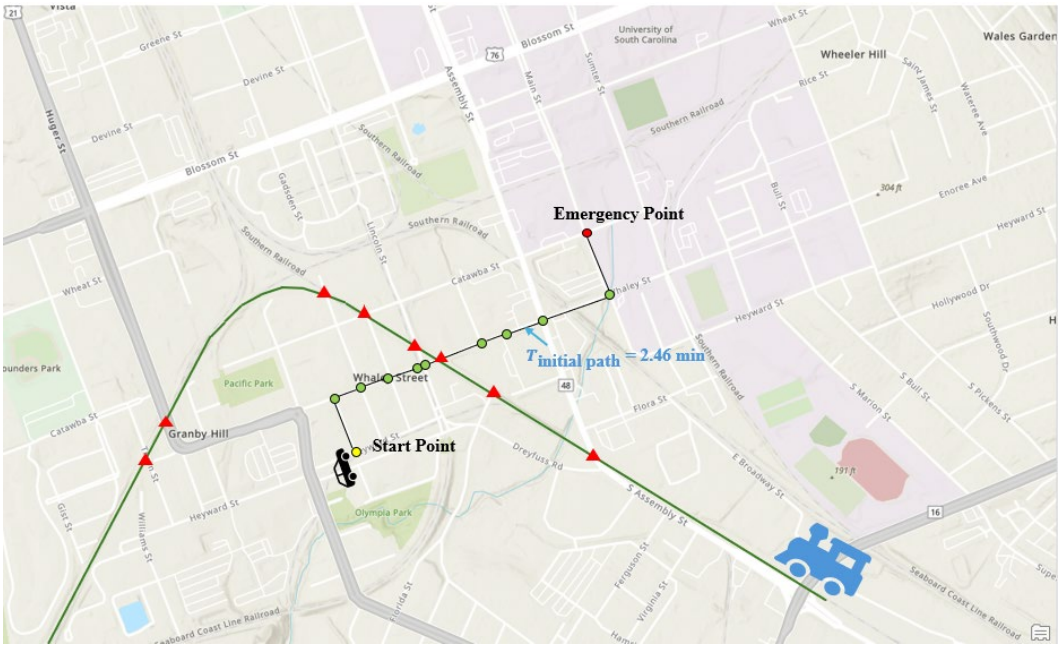


Figure 14 Initial Route Planning for Scenario 1

In the figure 14, the dark green trajectory represents the planned route of the train, while the yellow dot indicates the current location of the police officer, and the red dot marks the location of the emergency call. Green points represent intersections along the initially planned route, and red triangles denote level crossings that the train is expected to pass through. The black line corresponds to the initially generated route, with an estimated travel time of 2.46 minutes.

Notably, as the police officer proceeds along Whaley Street, it becomes evident that the route intersects with a level crossing at risk of being blocked by the oncoming train. This potential conflict necessitates dynamic path replanning to avoid unnecessary delays and ensure timely arrival at the emergency site.

We assume that as the officer proceeds along Whaley Street, they observe the train already passing through the level crossing, with 13 vehicles queuing ahead, indicating that the train did not just begin crossing but is already well into the blockage period. At this point, our decision-support system activates the Total Delay Time Estimation module, which models the potential delay introduced by the train as a probabilistic distribution. By performing Monte Carlo sampling and averaging the results, the system estimates the expected delay at this crossing to be approximately 8.93 minutes.

Consequently, if the officer were to continue along the originally planned route, the estimated arrival time would increase from 2.46 minutes to a total of 11.39 minutes, which is clearly unacceptable for an emergency response scenario. Therefore, the system initiates a real-time route replanning process to avoid the obstructed crossing. The newly optimized path is illustrated in the figure below.

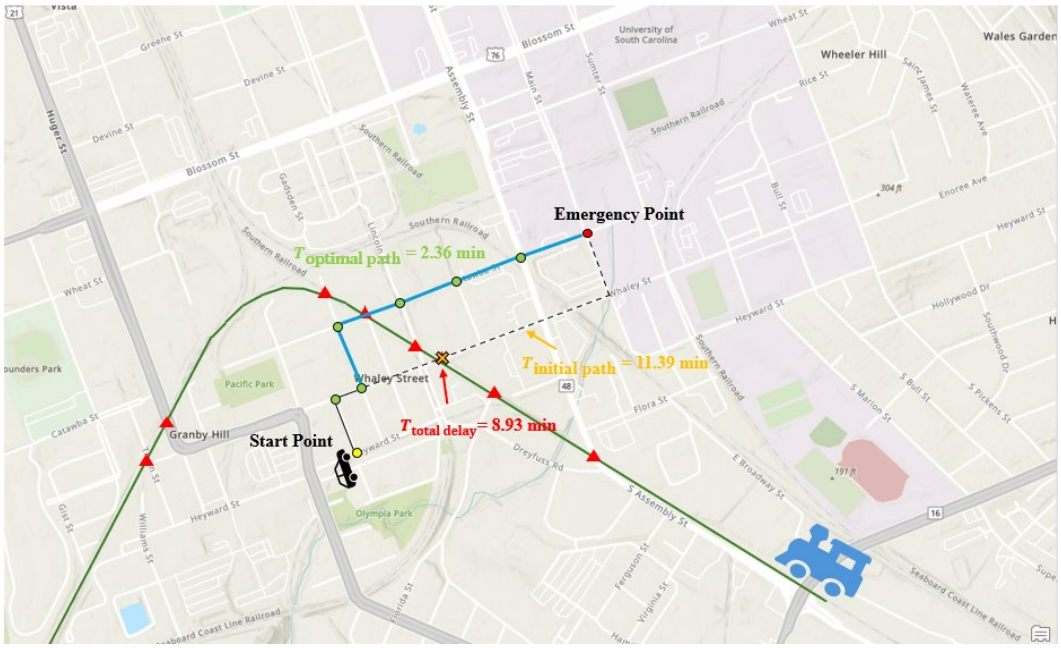


Figure 15 Route Replanning for Scenario 1

In figure 15, the bold purple line represents the replanned route. This new path directs the police vehicle to make a left turn at the intersection and proceed along Catawba Street. Although this alternative route also involves crossing a level crossing, the train has not yet arrived at that location, allowing uninterrupted passage. The total estimated travel time for the newly optimized route is 2.36 minutes. Compared to the 11.39 minutes required if the officer were to remain stationary and wait at the already-blocked Whaley Street crossing, the replanned route achieves a time saving of approximately 79.27%.

In the second scenario, we consider an emergency call occurring at 9:14 PM. At this moment, the police officer is patrolling a residential area near Gist Street, while the emergency is reported at the intersection of Richland Street and Assembly Street in downtown Columbia. The officer must immediately proceed to the incident location. Simultaneously, a train is traveling from the northwest corner of the city toward the southern direction.

Figure 16 Initial Route Planning for Scenario 2

We assume that the officer makes a right turn from Gist Street onto Laurel Street, where the train is observed to be passing through a level crossing. Due to the late evening hours, traffic is relatively light, with only two vehicles queued ahead. At this point, our route optimization system is activated. The Total Delay Time Estimation module, based on the joint sampling model, estimates the expected delay caused by the train to be approximately 2.56 minutes. This information is then passed to the route replanning module, which compares the projected delay of continuing along the current route with the travel time of potential alternative paths.

As illustrated in figure 17, the total delay time in this scenario is relatively short. This is likely because, by the time the officer observed the blocked level crossing on Laurel Street, the train was already nearing the end of its passage, thereby minimizing the duration of the blockage. Additionally, the light evening traffic, with only two vehicles ahead, resulted in a relatively short clearance time. The bold dark purple line represents the newly replanned route. Notably, the route planner did not choose the initially shorter path via the level crossing on Taylor Street, which may have also been blocked by the same train. Instead, the system selected a more distant alternative via Hampton Street.

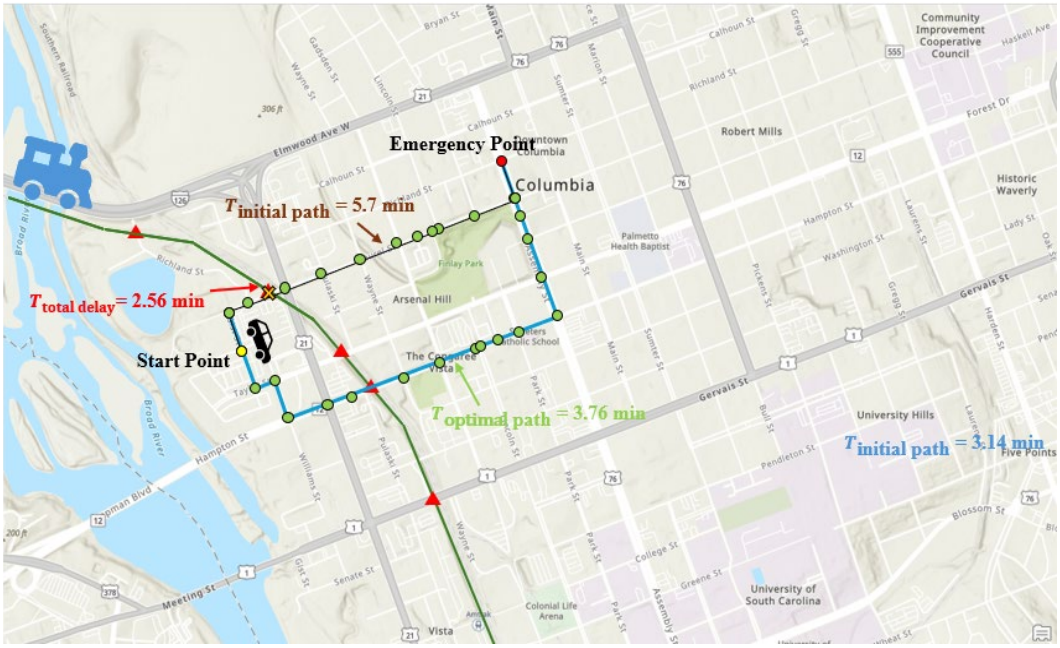


Figure 17 Route Replanning for Scenario 2

Although this detour results in a slightly longer travel time of 3.76 minutes, it still offers a time saving of 1.94 minutes compared to waiting at the blocked crossing. A comparison of the two routing options is summarized in table 2.

Table 2 Summary of Comparison of Path Choosing Strategies

	Initial Path Time Cost(min)	Estimated Total Delay Time(min)	Stop-and-Wait Strategy Time Cost(min)	Reroute Path Strategy Time Cost(min)
Scenario 1	2.46	8.93	11.39	2.36
Scenario 2	3.14	2.56	5.7	3.76

5 CONCLUSIONS

This study develops an intelligent path optimization system tailored for emergency response scenarios at railway level crossings. By integrating a joint probabilistic delay estimation model with a dynamic label correcting path search algorithm, the system effectively addresses the uncertainty caused by train-induced blockages. Unlike conventional models that consider only train occupancy time, this system further incorporates vehicle arrival and clearance time modeling, enabling a more realistic estimation of the *total delay time*, which is then used as the cost function in path planning.

In a case study set in Columbia, South Carolina, the system demonstrates its ability to dynamically detect blocked paths and replan routes in real time, significantly improving emergency response efficiency. In two representative emergency scenarios, the system reduces response time by up to 79.27% compared to the traditional "stop-and-wait" strategy, showcasing its strong delay-awareness and path optimization capability.

Despite its promising application potential, the current system assumes an ideal traffic environment, without accounting for more complex dynamic factors such as traffic signals or road closures. Future research could incorporate real-time traffic data and multi-source environmental constraints to enhance the practical adaptability of the route planning results. Overall, this study provides both a theoretical foundation and a technical pathway for developing more efficient and reliable urban emergency response support systems.

6 REFERENCES

[1] Z. Wang and S. Zlatanova, "Safe route determination for first responders in the presence of moving obstacles," *IEEE transactions on intelligent transportation systems,* vol. 21, no. 3, pp. 1044-1053, 2019.

[2] C. Wen *et al.*, "Train dispatching management with data-driven approaches: A comprehensive review and appraisal," *IEEE Access,* vol. 7, pp. 114547-114571, 2019.

[3] O. Frank, "Two-way traffic on a single line of railway," *Operations Research,* vol. 14, no. 5, pp. 801-811, 1966.

[4] E. Petersen, "Over-the-road transit time for a single track railway," *Transportation Science,* vol. 8, no. 1, pp. 65-74, 1974.

[5] A. Higgins and E. Kozan, "Modeling train delays in urban networks," *Transportation Science,* vol. 32, no. 4, pp. 346-357, 1998.

[6] E. Petersen and A. Taylor, "A structured model for rail line simulation and optimization," *Transportation Science,* vol. 16, no. 2, pp. 192-206, 1982.

[7] M. M. Dessouky and R. C. Leachman, "A simulation modeling methodology for analyzing large complex rail networks," *Simulation,* vol. 65, no. 2, pp. 131-142, 1995.

[8] M. Sonawane and C. Dhawale, "Evaluation and analysis of few parametric and nonparametric classification methods," in *2016 Second International Conference on Computational Intelligence & Communication Technology (CICT)*, 2016: IEEE, pp. 14-21.

[9] M. F. Gorman, "Statistical estimation of railroad congestion delay," *Transportation Research Part E: Logistics and Transportation Review,* vol. 45, no. 3, pp. 446-456, 2009.

[10] F. Corman and P. Kecman, "Stochastic prediction of train delays in real-time using Bayesian networks," *Transportation Research Part C: Emerging Technologies,* vol. 95, pp. 599-615, 2018.

[11] O. Fink, E. Zio, and U. Weidmann, "Predicting time series of railway speed restrictions with time-dependent machine learning techniques," *Expert Systems with Applications,* vol. 40, no. 15, pp. 6033-6040, 2013.

[12] Q. Bing, D. Qu, X. Chen, F. Pan, and J. Wei, "Arterial travel time estimation method using SCATS traffic data based on KNN-LSSVR model," *Advances in Mechanical Engineering,* vol. 11, no. 5, p. 1687814019841926, 2019.

[13] J. Wu *et al.*, "A hybrid LSTM-CPS approach for long-term prediction of train delays in multivariate time series," *Future transportation,* vol. 1, no. 3, pp. 765-776, 2021.

[14] C. S. Tan, R. Mohd-Mokhtar, and M. R. Arshad, "A comprehensive review of coverage path planning in robotics using classical and heuristic algorithms," *IEEE Access,* vol. 9, pp. 119310-119342, 2021.

[15] E. W. Dijkstra, E. W. Dijkstra, E. W. Dijkstra, E.-U. Informaticien, and E. W. Dijkstra, *A discipline of programming*. prentice-hall Englewood Cliffs, 1976.

[16] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE transactions on Systems Science and Cybernetics,* vol. 4, no. 2, pp. 100-107, 1968.

[17] R. Bellman, "On a routing problem," *Quarterly of applied mathematics,* vol. 16, no. 1, pp. 87-90, 1958.

[18] J. Mosely, "Asynchronous distributed flow control algorithms," Massachusetts Institute of Technology, 1984.