Measurement 280 (2026) 121830

Contents lists available at ScienceDirect

Measurement

journal homepage: www.elsevier.com/locate/measurement

Optimizing multi-layer perceptron for travel time reliability prediction: a comparative study of water cycle and moth flame algorithms

Navid Khorshidi [a], Soheil Rezashoar [b], Shahriar Afandizadeh Zargari [a], Hamid Mirzahossein [c,*]

[a] School of Civil Engineering, Iran University of Science and Technology, Tehran, Iran
[b] Department of Transportation Planning, Faculty of Engineering, Imam Khomeini International University, Qazvin, Iran
[c] Department of Civil – Transportation Planning, Imam Khomeini International University, Qazvin, Iran

ARTICLE INFO

Keywords:
Travel Time Index (TTI)
Travel Time Reliability (TTR)
Multi-Layer Perceptron (MLP)
Moth-Flame Optimization (MFO)
Water Cycle Algorithm (WCA)

ABSTRACT

This study compares the performance of the Water Cycle Algorithm (WCA) and Moth Flame Optimization (MFO) algorithms in optimizing Multilayer Perceptron Neural Networks (MLPs) for estimating the Travel Time Index (TTI) in Virginia's transportation network. The input features included average annual daily traffic per lane-mile (AADT/(Lane·mile)), total accident rate, link length, and precipitation. Results showed that WCA outperformed MFO, achieving a lower MAE objective function, which indicated higher accuracy. However, MFO was significantly faster, with an execution time of 1137.74 s (approximately 19 min) compared to 2364.08 s (approximately 39 min) for WCA. WCA also exhibited superior performance across multiple metrics, including MSE, RMSE, MAE, R2, MAPE, and NSE, for both training and test datasets. Notably, WCA achieved a training R2 of 0.64 and a test R2 of 0.40, compared to MFO's values of 0.58 and 0.38, respectively. Feature importance analysis highlighted AADT/Lane·mile (rank 1) and link length (rank 2) as the most critical variables, underscoring the importance of traffic volume and road segment length in travel time prediction. In addition, four other metaheuristic algorithms (HHO, AHO, SMA, and EO) were also evaluated, showing intermediate performance between WCA and MFO. Overall, WCA is the preferred choice for high-accuracy predictions, particularly in applications such as traffic management systems where precision is critical, whereas MFO, with faster execution, is more suitable for scenarios requiring quick computational results or time-sensitive predictions.

1. Introduction

The dynamic characteristics inherent in transportation systems require ongoing performance evaluation as a fundamental necessity. A range of metrics is employed for such assessment, with travel time and its variability regarded as among the most essential indicators. The accurate prediction of travel time and its fluctuations constitutes a critical determinant of efficiency in both freight and passenger transport operations [1]. Research indicates that traffic congestion is a primary factor contributing to travel time variability. Traffic congestion can be categorized into two distinct types. The first, known as recurrent congestion, arises when demand exceeds supply and is considered predictable due to its recurring nature. The second, termed non-recurrent congestion, results from inherently unpredictable events [2]. Seven factors contribute to non-recurrent congestion: adverse weather conditions, road maintenance operations, traffic accidents and incidents, demand fluctuations, special events, physical bottlenecks related to capacity, and traffic control equipment [1].

The assessment of transportation network performance, particularly in the context of factors contributing to travel time variability, has led to the conceptualization of travel time reliability. Significantly, the predictability of travel time and its fluctuations is frequently regarded as more critical than the duration of travel time itself by both practitioners and decision-makers [3]. The Federal Highway Administration (FHWA) emphasizes consistency across different days and time periods in its definition of travel time reliability, defining it as follows: *"The consistency or predictability of travel times across different time periods and days"* [4]. The National Cooperative Highway Research Program (NCHRP) Report 311 defines travel time reliability by examining it from three perspectives:

* Corresponding author.
E-mail addresses: navid.amoei.khorshidi@gmail.com (N. Khorshidi), soheilshoar@edu.ikiu.ac.ir (S. Rezashoar), zargari@iust.ac.ir (S.A. Zargari), mirzahossein@eng.ikiu.ac.ir (H. Mirzahossein).

https://doi.org/10.1016/j.measurement.2026.121830
Received 31 October 2025; Received in revised form 1 May 2026; Accepted 8 May 2026
Available online 10 May 2026

- "The discrepancy between expected and actual travel times caused by non-recurring congestion"
- "The variability in travel times observed over a large sample of daily trips"
- "The systemic impact of non-recurring congestion is quantified by the intensity, duration, and spatial extent of congestion events"[5].

The Strategic Highway Research Program (SHRP) reports examine travel time reliability from the following perspectives:

- "The probability of a non-failure over time"
- "The variability of travel times on an extended highway section over the course of six months to one year for different time slices of the day"
- "State of variation in expected (or repeated) travel times for a given facility or travel experience; the variation 24 in travel time for the same trip from day to day (same trip implies the same purpose, from the same origin, to the same destination, at the same time of the day, using the same mode, and by the same route)"[1].

The formal definition of travel time reliability represented a significant advancement in its incorporation into transportation system planning processes. However, the development of a quantitative index was essential to measure the economic implications of travel time uncertainty and to assess the effectiveness of corresponding management strategies [6,7].

The establishment of formal definitions for travel time reliability, particularly those emphasizing quantifiable attributes such as consistency, variability, probability, or specific operational impacts, facilitated the derivation of numerical indices by enabling systematic quantitative assessment. Moreover, given the inherent complexity of the underlying statistical processes and concepts, the development of clear and interpretable numerical indices was imperative [8].

Various numerical indices have been proposed to quantify travel time reliability. Statistical indices, such as standard deviation, coefficient of variation, percentiles, and skewness, fall into one category. A second category, performance-based indices, is derived from the statistical distribution of travel times and the relationships between its percentiles, including the Buffer Index, Travel Time Index (TTI), and Planning Time Index (PTI). This study focuses on the TTI [9,8]. The TTI, calculated as the ratio of actual average travel time to free-flow travel time, is a dimensionless metric commonly used to assess the impact of traffic congestion severity [10].

1.1. Contribution to measurement of TTI

This study relates to Measurement Science by addressing the quantification of travel time reliability, a complex and nonlinear metric. Conventional deterministic and linear models have known limitations in capturing such nonlinear behavior. Although machine learning methods have been applied to travel time prediction, the Multi-Layer Perceptron (MLP) has seen limited use in this domain, and systematic optimization of MLP weights and biases using metaheuristic algorithms for TTI estimation has not been previously reported.

A metaheuristic-optimized MLP framework is proposed. This framework is designed to balance predictive accuracy with computational efficiency, offering an alternative to deep learning architectures such as Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM), which typically require higher computational resources and offer less interpretability. Six metaheuristic algorithms are benchmarked using multiple performance metrics (MAE, MSE, RMSE, R^2, MAPE, and NSE). The results provide practical guidance for designing measurement systems where both accuracy and execution time are relevant, particularly for real-time Intelligent Transportation Systems (ITS).

The use of multiple evaluation metrics offers a broader perspective compared to single-metric validation. Feature importance analysis is also employed to identify key variables affecting TTI variability, which aligns with principles of Explainable AI (XAI). This approach may support more efficient sensor deployment and data collection strategies, and could be applicable to other measurement domains where high-dimensional data and resource constraints exist.

1.2. Study amis and key contributions

This study aims to:

- Develop an optimized ANN (specifically MLP) architecture framework for TTI prediction by integrating the MFO and WCA metaheuristics.
- Evaluate the comparative effectiveness of MFO versus WCA in enhancing MLP performance for transportation applications.
- Compare the performance of MFO and WCA with four other metaheuristic algorithms to assess alternative optimization strategies and explore their potential trade-offs between predictive accuracy and computational efficiency.
- Investigate the impact of key variables, including traffic density (AADT per lane-mile), crash rates (TCr), road length, and precipitation, on travel time variability.

Key contributions:

- **Novel Methodology:** Proposes a hybrid metaheuristic-MLP approach to address TTI prediction challenges under traffic and weather conditions.
- **Variable Selection:** Identifies and validates critical predictors to guide future data-driven traffic models.
- **Algorithmic Comparison:** Offers insights into the trade-offs between optimization techniques for MLP tuning.
- **Practical Relevance:** Provides a framework adaptable to ITS for real-time congestion management and safety improvements.

The novelty of this study lies in several aspects. Although metaheuristic optimization of neural networks has been studied, the systematic benchmarking of WCA and MFO for optimizing MLP weights and biases in TTI prediction has not yet been addressed. Furthermore, this study evaluates four additional metaheuristic algorithms, namely Harris Hawks Optimization (HHO), Archerfish Hunting Optimization (AHO), Slime Mould Algorithm (SMA), and Equilibrium Optimization (EO), providing a broader understanding of the trade-offs between predictive accuracy and computational efficiency. While recent trends have shifted toward deep learning architectures such as CNN and LSTM, the present study utilizes an optimized MLP to balance accuracy with model simplicity. Additionally, aligning with XAI frameworks, this work employs feature importance analysis to quantify the impact of key variables on the travel time index.

This study is structured to systematically address the research objectives:

Section 2 provides an overview of research on ANNs, MLPs, and metaheuristic methods, including MFO and WCA. Section 3 introduces the Virginia case study, outlining the key variables in the dataset. Section 4 elaborates on the MLP architecture, its optimization using MFO and WCA, and the evaluation metrics. Section 5 compares the performance of MLP models optimized by MFO and WCA using the dataset. Section 6 summarizes the effectiveness of these optimization techniques. Section 7 explores applications in transportation management and ethical considerations, offering a clear and thorough analysis. Section 8 addresses the limitations of the study. Section 9 suggests directions for future research.

2. Literature review

The significance of travel time and its variability commonly termed travel time reliability has garnered substantial scholarly attention. While precisely tracing the origins of modeling travel time variability proves challenging, research endeavors from the 1970s are widely regarded as among the earliest contributions. Notably, in 1968, Gaver proposed a departure time choice model that explicitly accounted for travel time variations, despite not employing the term "travel time reliability" at that time [11]. Richardson and Taylor investigated the relationship between traffic congestion and travel time variability in Australia, focusing on the distribution of travel times for regular commuters as key users of transportation systems [12]. Richardson and Taylor's efforts to establish a connection between travel time variability and user behavior were acknowledged by Prashker, culminating in a 1979 study examining the reliability of travel modes and its influence on travel behavior. In this work, Prashker contended that the reliability of travel modes constitutes the primary characteristic of a transportation system shaping passenger travel behavior [13]. Studies conducted during this period primarily relied on average or fixed travel time values, as travel time uncertainty and variability were not yet a primary concern in travel time modeling.

The incorporation of stochastic components, reflecting the inherently variable and dynamic nature of travel time, emerged in subsequent generations of modeling approaches. This paradigm shift from deterministic modeling, which relies on average and fixed values, to stochastic modeling, which accounts for inherent variability and uncertainty established travel time reliability as a critical component of transportation planning. One of the earliest studies in this vein was Senna's 1994 research, which employed econometric models to assess how the value of time varies in response to changes in travel time reliability[14]. Given the increasing significance of travel time reliability in transportation evaluation and planning, modeling methodologies have advanced considerably. Linear regression models, widely adopted for their interpretability and ease of application, have proven effective in analyzing travel time reliability. Numerous studies have employed such models; notably, Elefteriadou and Cui developed a linear regression framework incorporating variables including traffic congestion, weather conditions, work zones, and accidents [15]. Some studies, such as 16, focused on predicting the TTI using linear models for peak and off-peak periods. Other studies, such as CP and Karuppanagounder[17], employed linear regression with variables including traffic volume, road geometry, and travel time to predict TTI with high accuracy R^2. Although the aforementioned applications of linear regression models yielded acceptable results, their limitations, such as collinearity and interdependencies among predictors, led Afandizadeh et al. [18]to describe linear regression as an unsuitable method for predicting travel time reliability. Similarly, time-series models like ARIMA, used in Billings and Yang's study[19]demonstrated satisfactory performance in certain scenarios but exhibited significant performance declines when applied to real-world data.

In response to the inherent limitations of linear regression models, machine learning techniques have emerged as a viable and effective alternative for modeling travel time reliability. The proliferation and advancement of data-driven approaches, particularly with the integration of big data and machine learning methodologies, have substantially improved the predictive accuracy and robustness of travel time reliability estimation. A key advantage of machine learning models lies in their capacity to operate without reliance on the specific distributional assumptions underpinning conventional linear regression methods [20]. Furthermore, as the role of big data and real-time data in travel time reliability prediction became more prominent, machine learning models successfully adapted to these advancements.

Various machine learning methods have been employed by researchers to predict and model travel time reliability. Table 1, summarizes selected studies that utilized machine learning techniques in their

Table 1
Literature review of studies conducted in travel time reliability modeling.

Row	Study (authors, year)	Year	Modeling technique	Key performance metric	Reported value
1	[52]	2005	Artificial Neural Networks	MRE / SRE	Nearly unbiased estimation (MRE ≈ 0%) with SRE of 29%
2	[53]	2017	K-Means Clustering	Qualitative assessment	Demonstrated practically viable clustering outcomes
3	[54]	2019	Decision Tree	Qualitative assessment	Showed strong ability to recognize dominant traffic patterns
4	[55]	2019	Neural Network	Qualitative assessment	Delivered overall reliable and stable results
5	[18]	2021	Decision Tree	R^2 / MSE	Excellent training fit ($R^2 = 0.92$) that dropped to moderate levels on unseen data ($R^2 = 0.47$)
6	[18]	2021	Support Vector Regression	R^2 / MSE	Moderate training performance ($R^2 = 0.57$) with limited generalization ($R^2 = 0.34$)
7	[18]	2021	K-Nearest Neighbors	R^2 / MSE	Strong training accuracy ($R^2 = 0.89$) yet noticeable decline on test set ($R^2 = 0.66$)
8	[56]	2020	Neural Network	R^2 by lane configuration	Outstanding fit for both two-lane ($R^2 = 0.98$) and three-lane ($R^2 = 0.97$) scenarios
9	[57]	2022	Neural Network	MAPE range	Mean absolute percentage error stayed within 11–17.8% range
10	[58]	2023	Random Forest	MAPE range	Achieved notably low MAPE values ranging from 5.12% to 11.5%
11	[59]	2024	Random Forest	Prediction target	Targeted the central tendency (median) and upper tail (90th percentile) of travel times
12	[60]	2024	Bagging-based Regression	Best performing method	Gradient Boosting Regressor clearly outperformed other ensemble variants
13	[61]	2025	Random Forest	Prediction target	Focused on the full distribution of travel times, with particular strength at the median
14	[62]	2025	Neural Network	MAPE	Recorded an exceptionally low minimum MAPE of 1.14%, with 92% of predictions staying under 10% error

modeling frameworks. As the novelty of this research lies in its innovative methodology, the table focuses solely on reviewing and presenting the application of these methods, while efforts to introduce new data sources or variables are beyond the scope of this study. In the table, MRE stands for Mean Relative Error, SRE stands for Standardized Residual Error, and MAPE stands for Mean Absolute Percentage Error.

As shown in Table 1, ANN models are among the most widely used methods for modeling travel time reliability. Neural networks possess unique characteristics that make them well-suited for this purpose, including the ability to handle nonlinear relationships, robust performance with noisy data, advanced pattern recognition, and effective management of temporal and spatiotemporal complexities [21,22]. However, studies such as Afandizadeh Zargari et al. [18] reported that ANNs underperformed compared to other machine learning models, and optimizing hyperparameters is known to be a critical factor in enhancing their performance [23].

Existing approaches for travel time prediction, including conventional optimization algorithms such as Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) for optimizing neural network weights, suffer from several limitations. These include frequent convergence to suboptimal local minima, sensitivity to initial parameter settings, and lack of explicit optimization for reliability-based metrics such as the TTI. Furthermore, many existing methods do not specifically address the optimization of MLP weights and biases in the context of travel time reliability. The present study addresses these limitations by utilizing the WCA and MFO, which incorporate exploration–exploitation mechanisms for effective escape from local minima and reduced sensitivity to initial parameters, while formulating the objective function to minimize prediction error specifically for the TTI.

Recent trends have shifted toward hybrid models combining architectures such as CNN and LSTM. Despite their success, these deep learning models often act as “black boxes” and require extensive computational resources. In contrast, this study utilizes an optimized MLP to balance predictive accuracy with model simplicity, ensuring feasibility for real-time applications. Furthermore, aligning with XAI frameworks [24,25,26], feature importance analysis is employed to quantify the impact of key variables such as traffic density and precipitation, providing actionable insights into the drivers of travel time variability.

2.1. Research gap and relation to recent literature

An examination of the research background reveals that, despite the widespread use of ANNs in travel time reliability estimation, none of the studies reviewed have utilized the MLP [27]. Beyond the novel application of the MLP to this domain, the literature review indicates that metaheuristic algorithms have not yet been employed to optimize the MLP's structural parameters, specifically its weights and biases, for travel time reliability prediction. To address this, the present study utilizes the MFO algorithm and the WCA to optimize the MLP.

A focused review of the recent literature shows that, although several studies have advanced travel time prediction using hybrid machine learning approaches, the specific application of the MLP for TTI prediction has not been explored. Furthermore, no previous study has systematically benchmarked WCA and MFO for optimizing MLP parameters in this context. To provide a broader perspective, this study also includes a comparative evaluation of four additional recently developed metaheuristic algorithms. The prevailing trend in recent research has been dominated by deep learning architectures such as CNN and LSTM. Despite their demonstrated success, these models often function as “black boxes” and impose substantial computational demands. In contrast, the present study proposes an optimized MLP framework that offers a balance between predictive accuracy and computational efficiency, which may be particularly relevant for real-time ITS applications.

3. Case study and dataset collection

3.1. Case study

As highlighted in the literature review, traffic congestion contributes to travel time variability, thereby necessitating the definition and analysis of travel time reliability. The present study focuses on the interstate highway network in Virginia. Given the context of high-speed roadways, one of the seven factors commonly identified as contributing to congestion namely, traffic control devices was excluded from the modeling process. Furthermore, due to the annual resolution of the data, which is appropriate for long-term transportation planning, road maintenance operations and special events were also omitted. Accordingly, the variables incorporated in this study include adverse weather conditions, traffic accidents and incidents, demand fluctuations, and physical capacity bottlenecks.

This study utilized data from 59 highway segments within Virginia’s interstate highway network. Each segment was analyzed in both directions (outbound and return) over four consecutive years (2014–2017), yielding a total of 472 data records. Fig. 1 illustrates the spatial distribution of these segments across the state.

3.2. Dataset collection

To represent demand fluctuation and physical bottlenecks in the proposed models, the Average Annual Daily Traffic (AADT), number of lanes, and the length of each segment is considered. The State of Virginia AADT year books[1] were used to obtain the AADT data for the analysis period for the interstate locations for the 4-year analysis period. The AADTs covered the years 2014–2017. During the analysis period, the AADT was stable at the majority of the locations. The dataset did not include locations where AADT or roadway geometry changed unreasonably. The route segments between interchanges were selected. Google Maps was used to extract the number of lanes and the length of each road segment. Since data on queue length could not be collected, each location was examined separately. Although queue spillbacks have an impact on the reliability of travel times, this study has not taken that into account. This is because there is a lack of reliable data, and planning studies will not consider this data either. To simultaneously consider these three components, the study suggests AADT/(Lane.mile), a variable that includes AADT, length of segment, and its cross section.

The State of Virginia Crash Database provided the crash data.[2] Roadway linkages were used in ArcMap to connect the points that indicated crash locations. Using the AADT and the duration of each network link, the number of crashes of each type including property damage alone, injury, and fatal accidents were translated into crash rates. This variable is shown by TCr.

Also, three variables representing adverse weather conditions: the number of days with rainfall intensity less than 1 in., the number of days with rainfall intensity less than 0.1 in., and the annual precipitation in millimeters. During data preparation, the non-point-specific nature of rainfall (occurring over an area) was considered, and meteorological data for each highway segment were obtained from the nearest weather station. Statistical data for these variables were sourced from the National Oceanic and Atmospheric Administration (NOAA) website. The results of correlation shows that annual precipitation shown by PRCP has an acceptable level of correlation. Based on the values of correlation analysis, PRCP is chosen.

3.2.1. Correlation matrix analysis

Table 2 presents the correlation matrix illustrating the association

[1] https://www.virginiadot.org/info/ct-TrafficCounts.asp.

[2] https://public.tableau.com/views/Crashtools8_2/Main?%3Aembed=y&%3AshowVizHome=no&%3Adisplay_count=y&%3Adisplay_static_image=y.

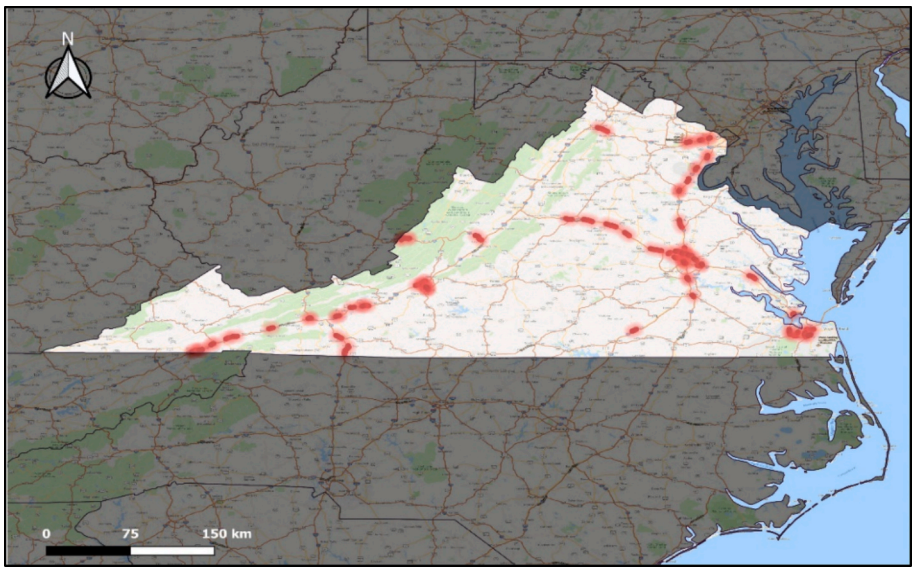

Fig. 1. Spatial location of segments in virginia interstate.

Table 2
Correlation matrix.

	$\frac{PRCP}{1000}$	Link Length	TCr	$\frac{AADT}{Lane.mile}$	TTI
$\frac{PRCP}{1000}$	1.00	−0.21	−0.08	0.18	−0.07
Link Length	−0.21	1.00	−0.07	−0.70	−0.15
TCr	−0.08	−0.07	1.00	0.18	0.56
$\frac{AADT}{Lane.mile}$	0.18	−0.70	0.18	1.00	0.32
TTI	−0.07	−0.15	0.56	0.32	1.00

between TTI as the dependent variable and the four explanatory variables, namely PRCP/1000, Link Length, TCr, and AADT/Lane.

The results indicate a very weak negative correlation (−0.07) between TTI and PRCP/1000, suggesting that precipitation exerts a negligible influence on travel time reliability within the context of the applied modeling framework. Similarly, a slight negative relationship (−0.15) is observed between TTI and Link Length, implying that longer roadway segments may be associated with a marginal reduction in TTI. This pattern may reflect differences in traffic flow characteristics or geometric and design features of the roadway.

Conversely, a relatively strong positive correlation (0.56) is found between TTI and TCr, highlighting the substantial role of this variable in explaining variations in travel time reliability. In addition, TTI shows a moderate positive association (0.32) with AADT/Lane, indicating that higher traffic volumes per lane tend to increase travel time variability, although the magnitude of this effect is less pronounced compared to that of TCr.

The data's statistical characteristics are shown in Table 3 which also includes the 25th, 50th, and 75th percentiles, the mean, standard deviation, minimum and maximum recorded values for each feature, and the total number of observations.

Table 4 shows an example of data prepared for modeling.

4. Research methodology

The study will focus on optimizing MLP neural networks using metaheuristic algorithms within the research methodology. It will describe the MLP architecture, detailing its structure as the core ANN model. The MFO algorithm will be explored for its mechanism in optimizing MLP weights and biases, while the WCA will be examined for its nature-inspired approach to enhance MLP performance. A hybrid approach combining MFO and WCA will be developed to improve the MLP's effectiveness. Model performance will be evaluated using metrics such as MSE, and other relevant indicators.

The structure of the MLP is discussed in detail below.

4.1. Multi-Layer Perceptron (MLP)

The architecture of a MLP comprises three fundamental layers: an input layer responsible for receiving raw data, with its node count corresponding to the number of data features; a hidden layer where primary

Table 4
Sample data.

TTI	$\frac{AADT}{Lane.mile}$	TCr	Link Length	$\frac{PRCP}{1000}$
1.00	0.422139	0.166771	5.33	0.8389
1.03	0.440901	0.199593	5.33	1.1905
1.00	0.459662	0.382892	5.33	1.0971
1.01	0.469043	0.375235	5.33	0.9004
1.04	0.444664	0.087835	5.06	0.8389

Table 3
Statistical details of the data.

	Count	Mean	Std	Min	25%	50%	75%	Max
TTI	472.00	1.01	0.08	0.89	0.98	1.00	1.01	1.67
$\frac{PRCP}{1000}$	472.00	1.19	0.20	0.77	1.04	1.16	1.32	2.04
Link Length	472.00	2.76	1.64	0.66	1.64	2.32	3.53	7.92
Total Rate	472.00	0.17	0.12	0.02	0.09	0.14	0.21	0.87
$\frac{AADT}{Lane.mile}$	472.00	6.92	5.20	0.37	2.72	5.43	10.08	28.05

data processing occurs; and finally an output layer that delivers the final result. Each neuron in the hidden layer receives inputs from the previous layer's neurons and transmits outputs to the subsequent layer, incorporating nonlinear activation functions. The output layer generates the network's final output, which may consist of one or more neurons depending on the problem's requirements. The literature review has shown the widespread application of MLP in various fields, such as emission prediction [28] and speed control [29].

The operation of an MLP can be described in four key stages [30,27]:

- **Forward Propagation:** Each neuron multiplies the received inputs by their corresponding weights, adds a bias, and passes the weighted sum through a nonlinear activation function to produce the neuron's output.
- **Cost Function:** After generating the output, a cost function evaluates the prediction error.
- **Backpropagation:** To minimize the cost function, the backpropagation algorithm calculates gradients to adjust the network's weights and biases.
- **Optimization:** Using gradients and an optimization algorithm, the weights and biases are iteratively updated to reduce the error.

As previously indicated, the novelty of this research lies in the optimization of the MLP's weights and biases using two metaheuristic algorithms: MFO and WCA. Within the MLP architecture, weights and biases perform essential functions in processing input data and generating outputs. Weights determine the strength of connections between neurons, whereas biases serve as activation thresholds, enabling the network to detect patterns independently of the input values alone. The precise optimization of these parameters enhances pattern recognition capabilities, reduces prediction errors, and improves overall accuracy, generalizability, convergence speed, and computational efficiency.

Conventional MLP training methods, such as backpropagation, which rely on gradient-based techniques, may face challenges in complex, non-convex search spaces, including the risk of getting trapped in local minima. These limitations prompted the adoption of metaheuristic optimization algorithms, which can more effectively explore the search space and identify solutions close to the global optimum.

The structures of the two aforementioned algorithms, MFO and the WCA, will be detailed in the subsequent sections.

4.2. Moth-Flame Optimization (MFO)

The Moth-Flame Optimization (MFO) algorithm, a relatively recent and efficient metaheuristic introduced by Mirjalili in 2015, is inspired by the nocturnal navigation behavior of moths. Its novelty lies in modeling a unique navigation mechanism known as transverse orientation. In nature, moths maintain a fixed angle relative to a distant natural light source, such as the moon, enabling them to fly in straight lines over long distances. However, when exposed to an artificial light source, such as a flame, this navigation mechanism is disrupted, causing moths to spiral toward the light. This spiraling trajectory serves as the conceptual basis for convergence toward optimal solutions within the search space. The MFO algorithm comprises two primary components: moths and flames. Each moth represents a candidate solution to the optimization problem, while each flame denotes the best positions i.e., optimal solutions identified by the moths up to the current iteration. The navigation mechanism is governed by moths moving in a spiral pattern around flames, which facilitates a balance between exploration where moths search the solution space broadly and exploitation where moths converge toward the most promising flames. The algorithm proceeds through three key stages: initialization, iterative update, and termination upon meeting a predefined stopping criterion. Notably, MFO operates with fewer parameters than many other metaheuristics, maintaining simplicity while effectively balancing exploration and exploitation, thereby enhancing its capacity to escape local minima [31]. The MFO could prove itself as a reliable optimization tool in various fields, such as training multi-layer perceptron classifier, [32], medical diagnosis [33]and managing smart cities [34].

4.3. Water Cycle Algorithm (WCA)

The Water Cycle Algorithm (WCA), inspired by the natural hydrological cycle, is a metaheuristic optimization algorithm designed to identify optimal solutions within a given search space. Its structure comprises three primary components: the sea, which represents the best solution discovered within the population; rivers, which correspond to a subset of the next best water droplets moving toward the sea; and streams, which denote the remaining droplets with comparatively lower fitness values. The WCA operates through five key stages, systematically guiding the population toward convergence on the global optimum [35,36]:

- **Initialization:** A population of water droplets is randomly generated in the search space, and based on their fitness with respect to the objective function, the best droplet is designated as the sea, the next best as rivers, and the remaining as streams.
- **Movement:** Streams flow toward rivers or the sea, and rivers flow toward the sea, modeling the convergence toward better solutions.
- **Evaporation and Precipitation:** To avoid local optima and enhance diversity, an evaporation and precipitation process is applied.
- **Solution Update:** The best solution is updated in each iteration.
- **Termination:** The process continues until the stopping criterion (or criteria) is met.

4.4. Hybrid MLP optimization with metaheuristic algorithms

ANNs are composed of sequential layers, each comprising multiple computational units referred to as neurons. These units are equipped with adaptable activation functions, the weight parameters of which are calibrated using preexisting information and domain-specific expertise. The overall architecture of an ANN consists of multiple hierarchical levels, encompassing the fundamental input and output layers, as well as intermediate hidden layers that are incorporated as necessary between these two primary components [37].

The computation of the weighted aggregation of input values begins with Equation (1). Here, n indicates the overall count of input neurons within the MLP structure, while h refers to the number of neurons in the hidden layer. The term w_{ij} corresponds to the weight that links the *ith* input variable X_i to the *jth* hidden neuron, and β_j represents the bias associated with that same hidden neuron, where j takes values from 1 to n [38]:

$$S_j = \sum_{i=1}^{n} w_{ij}.X_i + \beta_j \tag{1}$$

Following the computation performed by the aggregation function presented in Eq. (1), an activation mechanism is subsequently triggered to generate the neuronal output. A range of activation functions may be adopted within MLP architectures; in this study, the sigmoid function frequently employed in earlier MLP-based research is selected. This function serves to map the weighted output from the hidden neurons to the succeeding layer. The output produced by the *jth* node in the hidden layer is obtained via Eq. (2), where S_j denotes the aggregated weighted input reaching that node, and f represents the sigmoidal activation function applied at the *jth* hidden neuron [38]:

$$f_j = \frac{1}{1 + e^{-S_j}} \tag{2}$$

The ultimate stage of the MLP computation process consists of deriving the network's final output through Equation (3). This calculation is performed subsequent to obtaining the activation values from

each neuron positioned within the hidden layer. In this expression, β_k corresponds to the bias component associated with the *kth* output neuron, while w_{kj} indicates the weight coefficient that links the *jth* hidden unit to the *kth* output node [38]:

$$\widehat{y}_k = \sum_{i=1}^{m} w_{kj} f_i + \beta_k \tag{3}$$

The final output generated by the configured MLP architecture is fundamentally influenced by its weight and bias components, as formally expressed in Eqs. (4)–(6). Optimizing these parameters by identifying suitable values constitutes a pivotal step in improving the learning process of MLP networks [38].

For metaheuristic methods to be successfully applied in training MLP networks, a proper formulation of the optimization task is essential. The structure of the MLP is defined based on the characteristics of the dataset and architectural choices, specifically the number of hidden neurons (h), the input feature dimension (f), the total weight count (w), the output dimensionality (o), and the number of bias terms (b). These quantities are derived through the following expressions, which determine the total number of weights and biases:

$$h = (2 \times f) + 1 \tag{4}$$

$$w = (f \times h) + (h \times o) \tag{5}$$

$$b = h + o \tag{6}$$

Within the framework of optimizing MLP networks through metaheuristic techniques, the primary objective is typically to reduce an error measure that quantifies the discrepancy between estimated and observed outputs. For this investigation, the Mean Absolute Error (MAE) is adopted as the fitness function to be minimized during the optimization phase. The adjustment of weight coefficients and bias parameters within the MLP structure is carried out with the explicit purpose of lowering this MAE value. The architectural design of the MLP comprises several distinct levels: an input layer, one or more hidden layers, and an output layer, where each neuron across these levels is interconnected via weighted connections. These weight parameters, together with the bias components, undergo iterative refinement throughout the learning process, with MAE serving as the guiding criterion. The metaheuristic search algorithms update these decision variables at each iteration to progressively reduce the error. MAE itself is computed as the mean of the absolute deviations between predicted and ground truth values, mathematically expressed in the following form:

$$MAE = \frac{1}{n} \sum_{n=1}^{n} \left(\frac{y_i - \widehat{y}_i}{n} \right) \tag{7}$$

n is the predicted value for each data point. y_i is the actual value, and $\widehat{y}_i$ is the predicted value for each data point. MAE offers an intuitive measure of prediction error by averaging the absolute discrepancies without considering whether the deviation is positive or negative. This metric is especially advantageous when substantial divergences from true values are considered critical. A smaller MAE reflects superior model performance, indicating that the predicted outcomes lie in closer proximity to the actual observations. Such characteristics render MAE a favorable evaluation criterion in regression-oriented tasks such as MLP optimization, particularly owing to its reduced sensitivity to extreme values relative to alternative performance measures.

The key hyperparameters of the MLP architecture and the metaheuristic optimization process are summarized in Table 5, together with the rationale for the selected values based on preliminary trials and common practices in similar transportation prediction studies.

Both the MLP and the metaheuristic algorithms (MFO and WCA) involve various configurations and parameters. The final hyperparameter values were determined based on the experiments conducted and results presented in Table 5. These selected values are

Table 5
Key hyperparameters, selected values, and selection rationale.

Hyperparameter	Values considered/ tested (if any)	Observed impact on performance (MAE or convergence)	Rationale/ justification for final choice
Number of neurons in hidden layer	8, 10, 12, 16, 20 (preliminary)	Best or near-best performance around 10	Good balance between model capacity and risk of overfitting given the limited dataset size (472 samples) and low number of input features (4)
Number of hidden layers	1, 2	1 layer showed better or comparable results	With only 4 input features and moderate sample size, adding a second layer did not provide meaningful improvement and increased complexity
Hidden layer activation function	—	—	tansig (hyperbolic tangent sigmoid) is widely used and performs well in regression problems with bounded non-linear behavior
Output layer activation function	—	—	purelin (linear) is the standard choice for regression tasks (predicting continuous TTI values) as it imposes no range restriction on the output
Population size (MFO and WCA)	30, 50, 80, 100, 150 (preliminary)	Good and stable convergence at 100	Provides sufficient search diversity while keeping computational time reasonable; common in metaheuristic-based neural network optimization
Maximum number of iterations	300, 500, 700, 1000 (preliminary)	Stable convergence typically reached around 400–450	Little additional improvement observed after 500 iterations; this value keeps total runtime acceptable

Table 6
Configuration parameters of MLP and metaheuristic algorithms (MFO & WCA).

Component	Parameter	Value/setting	Description
MLP	Data Split Ratio	75% training, 25% testing	Ensures sufficient data for both training and validation.
	Hidden Layer Size	10 neurons	Single hidden layer configuration.
	Activation Functions	'tansig' (hidden), 'purelin' (output)	Combines nonlinear and linear transformations.
Metaheuristics	Population Size (N)	100 agents	Balances exploration and computational efficiency.
	Maximum Iterations	500	Allows full convergence without excessive computation.
	Search Space Bounds	[−1, 1]	Optimal range for neural network weight optimization.
	Algorithms	MFO and WCA	Implemented with identical parameters for fair comparison.

comprehensively summarized in Table 6 and are described in more detail below.

The implementation follows a systematic approach as described below:

- **Network Configuration and Initial Setup:** In this study, the ANN was designed with a single hidden layer comprising 10 nodes. The activation function applied in the hidden layer is tan-sigmoid (tansig), while the output layer employs a linear (purelin) activation function. The rationale behind choosing 10 neurons for the hidden layer is to achieve equilibrium between model complexity and computational performance, allowing the network to effectively learn patterns inherent in the data. The choice of 10 neurons was made because using more or fewer neurons did not result in significant improvements or changes in the outcomes. The tan-sigmoid function is selected for the hidden layer due to its ability to introduce non-linearity, enabling the network to model more complex relationships. For the output layer, a linear activation function is used, which is standard for regression tasks, as it allows the model to predict continuous, unbounded values without any restrictions on the output range. To properly validate the model, the dataset is split into two subsets: 75% for training and 25% for testing. This division ensures that the model has a sufficiently large dataset to learn meaningful patterns, while retaining an adequate portion of the data for testing, thus evaluating the model's ability to generalize to new, unseen data.
- **Metaheuristic Integration and Optimization Process:** Both MFO and WCA algorithms are implemented with identical parameters (100 search agents, 500 maximum iterations, and [-1, 1] search bounds) to enable fair performance comparison. In the WCA and MFO algorithms, the optimization of weights and biases of a neural network is performed iteratively, relying on the interaction between an initial population and the number of algorithm iterations. This process begins with the random generation of a population of solutions, where each individual represents a unique set of initial values for the weights and biases of the neural network, serving as the starting point for the optimization search. Each candidate solution is evaluated using the MAE calculation as a minimization-type objective function. Throughout each iteration, the algorithms refine these weights and biases by exploring the solution space, drawing on distinct principles: WCA simulates water droplets moving through the space, guided by various forces to identify optimal positions, while MFO mimics the movement of moths toward flames to search for the best solutions.
- **Performance Evaluation:** The optimized weight configurations undergo rigorous validation using a comprehensive suite of evaluation metrics: MSE measures overall prediction variance, RMSE provides error magnitude in the target variable's units, while MAE and MAPE offer robust, scale-independent measures of accuracy. The R2 quantifies variance explanation capability, and NSE assesses model performance relative to observed mean. This multi-criteria assessment delivers robust insights into prediction accuracy (MSE, RMSE), practical error magnitude (MAE, MAPE), explanatory power (R2), and model reliability (NSE).

4.5. Model performance evaluation metrics

The research employed seven key metrics to assess model performance. R2 measures how well independent variables explain the dependent variable's variance, ranging from 0 to 1 with higher values indicating better fit. However, R2 naturally increases with additional predictors regardless of actual predictive improvement, necessitating the adjusted R2 calculation that accounts for predictor count and sample size[39]. MSE represents the average squared prediction errors, sensitive to large deviations due to squaring. Its square root yields RMSE, a scale-dependent metric analogous to standard deviation. MAE computes the average absolute error, directly reflecting error magnitude. MAPE converts MAE to percentage terms, offering intuitive 0–100% interpretation with greater outlier resistance than RMSE/MSE. For all error metrics (RMSE, MSE, MAPE), lower values correspond to better model accuracy [39]. The NSE coefficient, which ranges from negative infinity to 1 (with 1 representing optimal performance), serves as a key metric for evaluating model effectiveness. An NSE value of 1 indicates perfect agreement between predicted and observed values, while negative values signify that the model performs worse than simply using the mean of the observed data as a predictor [40].

Table 7 presents a summary of model performance evaluation metrics.

5. Numerical results and analysis

This study was conducted using MATLAB 2023b on a system with an Intel® Xeon® E7-8890 v4 processor (2.20 GHz), 32 GB RAM, and Microsoft Windows 10 Pro (Version 22H2, build 19045.3324). All numerical modeling was performed on this platform.

The comparative performance of the MFO and WCA algorithms is presented in Tables 8 and 9 which report the objective function values and evaluation metrics, respectively. The comparison is conducted using two criteria: the best obtained MAE value and the execution time measured in seconds. According to the results, WCA achieves a lower MAE value of 0.026006 compared to 0.030001 for MFO, indicating that WCA provides higher predictive accuracy.

In contrast, MFO demonstrates a clear advantage in terms of computational efficiency. Specifically, MFO completes the optimization process in 1137.74 s (approximately 19 min), whereas WCA requires 2364.08 s (approximately 39.4 min), making WCA's execution time about 2.08 times longer than that of MFO. This observation highlights MFO's superior performance in terms of speed and computational cost. It should be noted that the reported execution time corresponds to the total duration of optimization and MLP training over 500 iterations.

Performance was evaluated using multiple metrics (MSE, RMSE, MAE, R2, MAPE, and NSE) on training and test datasets. WCA consistently outperformed MFO across all metrics. On the training set, WCA recorded lower MSE (0.002529 vs. 0.0028673), RMSE (0.050289 vs. 0.053547), and MAE (0.026006 vs. 0.030001), reflecting a better fit to the data. It's higher R2 (0.64235 vs. 0.5818) and NSE (0.62974 vs. 0.58021), along with a lower MAPE (10.15 vs. 12.01), indicate stronger explanatory power and reduced percentage errors. On the test set, WCA maintained its advantage with lower MSE (0.0027232 vs. 0.0029091), RMSE (0.052184 vs. 0.053936), and MAE (0.03062 vs. 0.031123), alongside higher R2 (0.40832 vs. 0.38401) and NSE (0.3985 vs. 0.35742). However, both algorithms exhibited a performance drop from training to test sets, with MFO showing a more pronounced decline in R2 (0.5818 to 0.38401) and NSE (0.58021 to 0.35742), suggesting WCA's greater robustness to unseen data. The higher MAPE on the test set (WCA: 11.476, MFO: 12.078) indicates generalization challenges, with

Table 7
Summary of model performance evaluation metrics.

Metric	Description	Range/ units	Interpretation
R^2	Measures variance explained by independent variables	0–1	Higher = better fit
MSE	Average squared prediction errors	≥0	Lower = better accuracy
RMSE	Square root of MSE, scale-dependent	≥0	Lower = better accuracy
MAE	Average absolute error	≥0	Lower = better accuracy
MAPE	MAE in percentage terms	0–100%	Lower = better accuracy
NSE	Nash-Sutcliffe efficiency coefficient	−∞ to 1	1 = perfect, <0 = worse than mean

Table 8
Objective function values and optimization runtime for MFO and WCA.

Algorithm	Optimal value (MAE Metric)	Optimization runtime (Seconds)
MFO	0.030001	1137.74
WCA	0.026006	2364.08

Table 9
Performance metrics.

Algorithm	Metric	Train Set	Test Set
MFO	MSE	0.0028673	0.0029091
	RMSE	0.053547	0.053936
	MAE	0.030001	0.031123
	R^2	0.5818	0.38401
	MAPE	12.01	12.078
	NSE	0.58021	0.35742
WCA	MSE	0.002529	0.0027232
	RMSE	0.050289	0.052184
	MAE	0.026006	0.03062
	R^2	0.64235	0.40832
	MAPE	10.15	11.476
	NSE	0.62974	0.3985

WCA being more accurate.

Overall, these results clearly demonstrate a trade-off between prediction accuracy and execution time. Although WCA achieves higher accuracy, this improvement comes at the cost of significantly longer computation time compared to MFO. Specifically, while WCA is preferable in applications where prediction accuracy is the primary objective and higher computational cost is acceptable, MFO offers a much faster alternative and is therefore more suitable for time-sensitive scenarios or situations with computational resource constraints. Consequently, the choice between WCA and MFO should be made based on the specific requirements and limitations of the intended application. It should also be noted that in interpreting the results of this study, when referring to the WCA algorithm as the superior method, the emphasis has been placed on accuracy metrics.

The observed differences in performance between the training and test datasets, particularly in terms of R2 (dropping from 0.642 to 0.408 for WCA and from 0.582 to 0.384 for MFO) and NSE (decreasing from 0.630 to 0.399 for WCA and from 0.580 to 0.357 for MFO), indicate a reduction in the model's generalization when applied to unseen data. Such behavior is commonly reported in neural network-based models, especially when they are trained on datasets with limited size.

The substantial gap between training and testing performance suggests that both models may be affected by a certain degree of overfitting. Several factors could contribute to this behavior. The dataset consists of 472 observations with only four input variables, which may have limited the amount of information available for learning complex nonlinear relationships. It should be noted that applying various hyperparameter settings for the MLP network in this study did not have a significant effect in controlling overfitting. Despite these limitations, the test set results particularly for the WCA-optimized model ($R^2 \approx 0.41$ and MAPE $\approx 11.5\%$) show that the models are still capable of capturing meaningful patterns in the data.

Nevertheless, these findings highlight the importance of incorporating additional regularization mechanisms and larger datasets (e.g., through data augmentation techniques) in future studies to further improve generalization performance.

The convergence trajectories of the WCA and the MFO algorithm, as depicted in Fig. 2 illustrate the evolution of the objective function value over 500 iterations. The WCA commences with an initial objective function value of approximately 0.044 and exhibits a pronounced decline within the first 50 iterations, reaching roughly 0.03. This is followed by a more gradual reduction, culminating in convergence to approximately 0.026 by the terminal iteration. This pattern indicates rapid initial convergence and the capacity to identify near-optimal solutions in the early stages; however, the subsequent gradual decline suggests diminishing returns in improvement and convergence toward either a local or global optimum. In contrast, the MFO algorithm initiates with an objective function value of approximately 0.09 and undergoes a steep decrease during the initial phase, falling below 0.06 by approximately 100 iterations. Thereafter, it converges more slowly to approximately 0.030 by the end of 500 iterations. This trajectory reflects rapid early convergence followed by incremental improvements with increasing iterations. The deceleration in the rate of improvement after 100 iterations similarly indicates limited further gains and convergence toward an acceptable optimum. Overall, the comparative analysis suggests that the WCA demonstrates superior performance in terms of final solution accuracy.

The performance evaluation of the MFO in predicting the TTI is comprehensively illustrated in Figs. 3, 4, and 5.

The regression charts (Training Data Regression with $R = 0.7628$, R^2

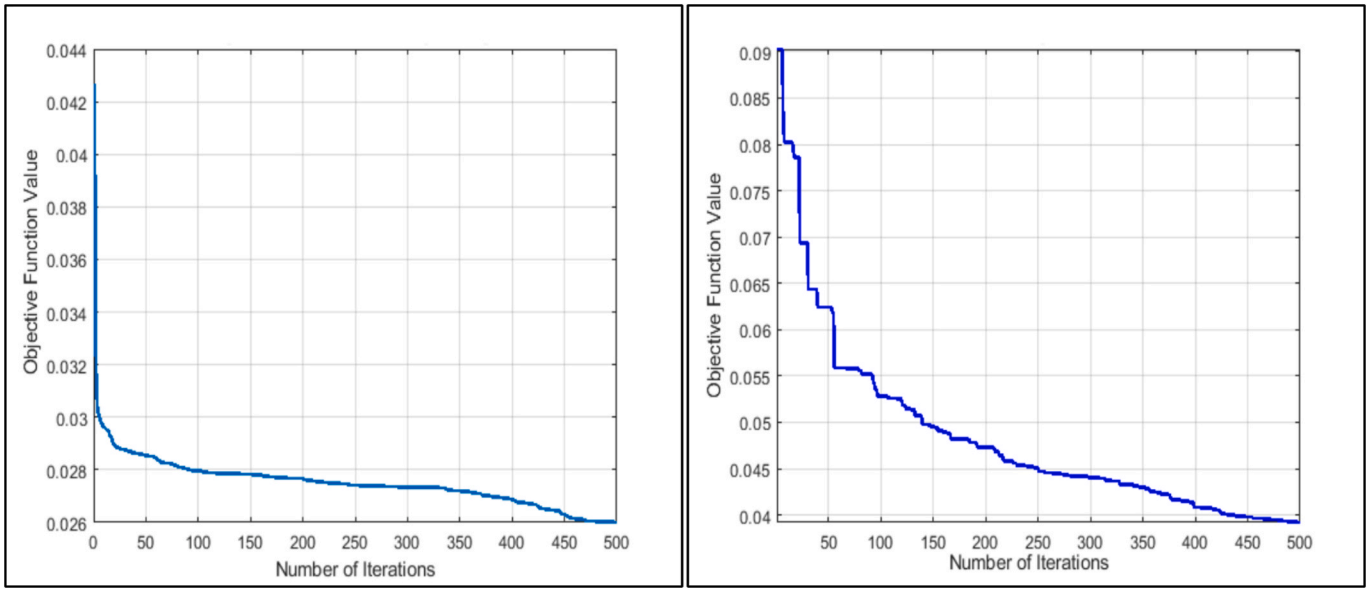

Fig. 2. Convergence curve plots (Left: WCA, Right: MFO).

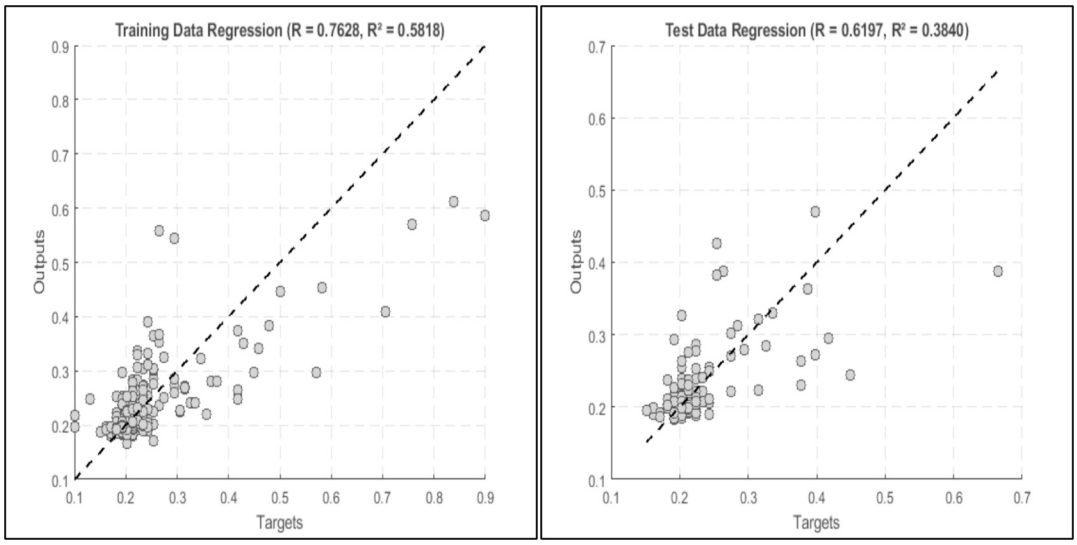

Fig. 3. MFO regression plots (left: training, right: testing).

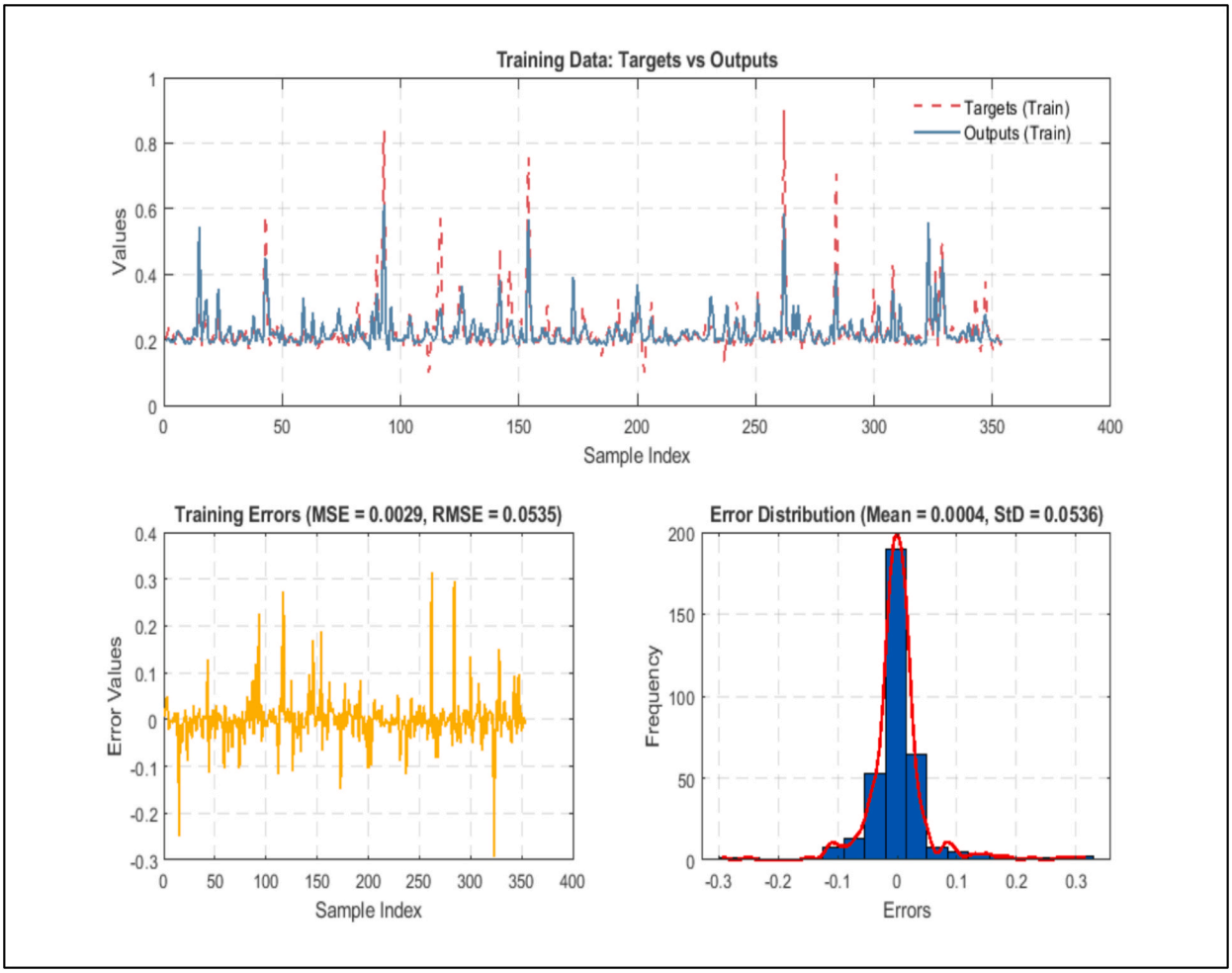

Fig. 4. Plots of MFO actual vs predicted (top), training errors (bottom left) and error distribution (bottom right).

= 0.5818 and Test Data Regression with R = 0.6197, R^2 = 0.3840) indicate a moderate to strong correlation between Targets and Outputs, though the drop in R^2 for test data suggests potential overfitting or differences in data distribution. The charts show significant fluctuations, yet Outputs generally align with Targets. The training errors (MSE = 0.0029, RMSE = 0.0535) and test errors (MSE = 0.0029, RMSE = 0.0539) are notably low, with error distributions centered on zero and a low standard deviation (STD around 0.05), confirming the model's good performance. However, the scatter of points and some inconsistencies suggest that the model may require adjustment or advanced techniques

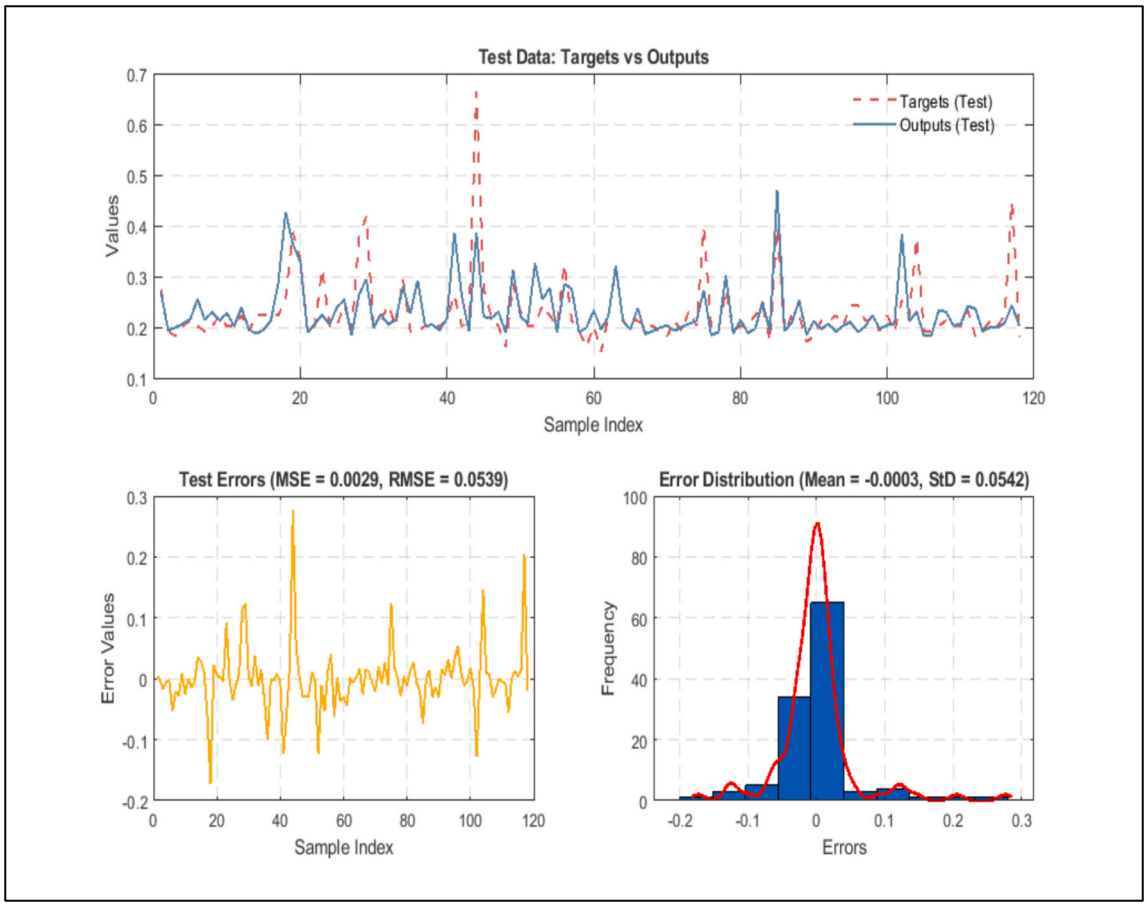

Fig. 5. Plots of MFO: actual vs predicted (top), testing errors (bottom left) and error distribution (bottom right).

when dealing with noisy or nonlinear data.

The performance evaluation of the WCA algorithm in predicting the TTI is comprehensively illustrated in Figs. 6, 7, and 8.

The regression charts (Training Data Regression with R = 0.8015, R^2 = 0.6423 and Test Data Regression with R = 0.6390, R^2 = 0.4083) indicate a strong correlation between Targets and Outputs in training data, which weakens slightly in test data, suggesting a moderate fit and potential overfitting or distribution differences. The charts reveal significant fluctuations in both Targets and Outputs (ranging from 0 to 1 for training and 0 to 0.7 for test), with Outputs generally aligning with Targets but showing some deviations. The error analyses (Training Errors: MSE = 0.0025, RMSE = 0.0503; Test Errors: MSE = 0.0027, RMSE = 0.0522) demonstrate low error rates, with error distributions centered on zero (mean ≈ 0.004–0.0067, STD ≈ 0.05) and near-normal shapes, indicating accurate predictions with random variations. Overall, the model performs well, though the drop in R^2 and observed fluctuations

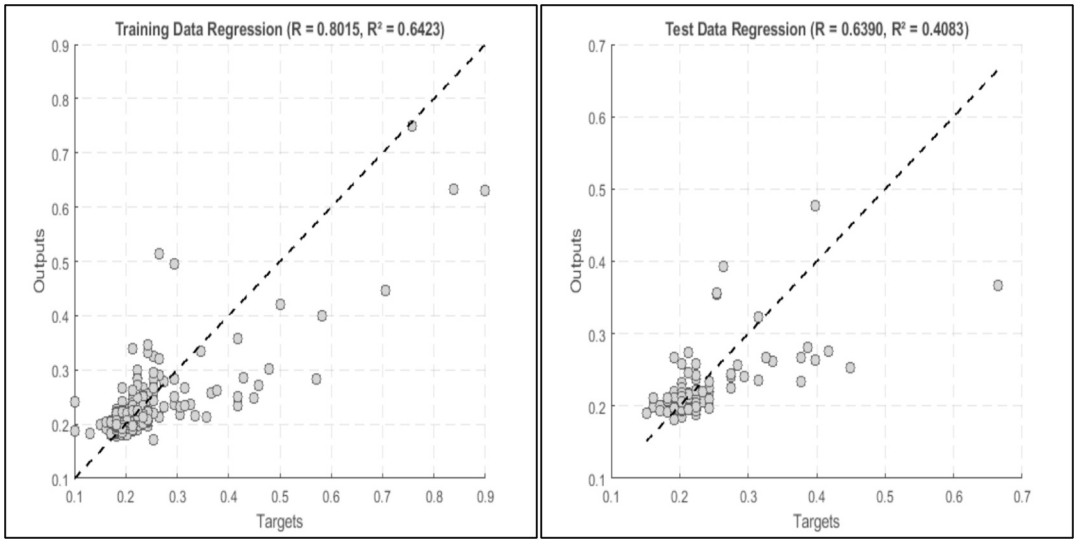

Fig. 6. WCA regression plots (left: training, right: testing).

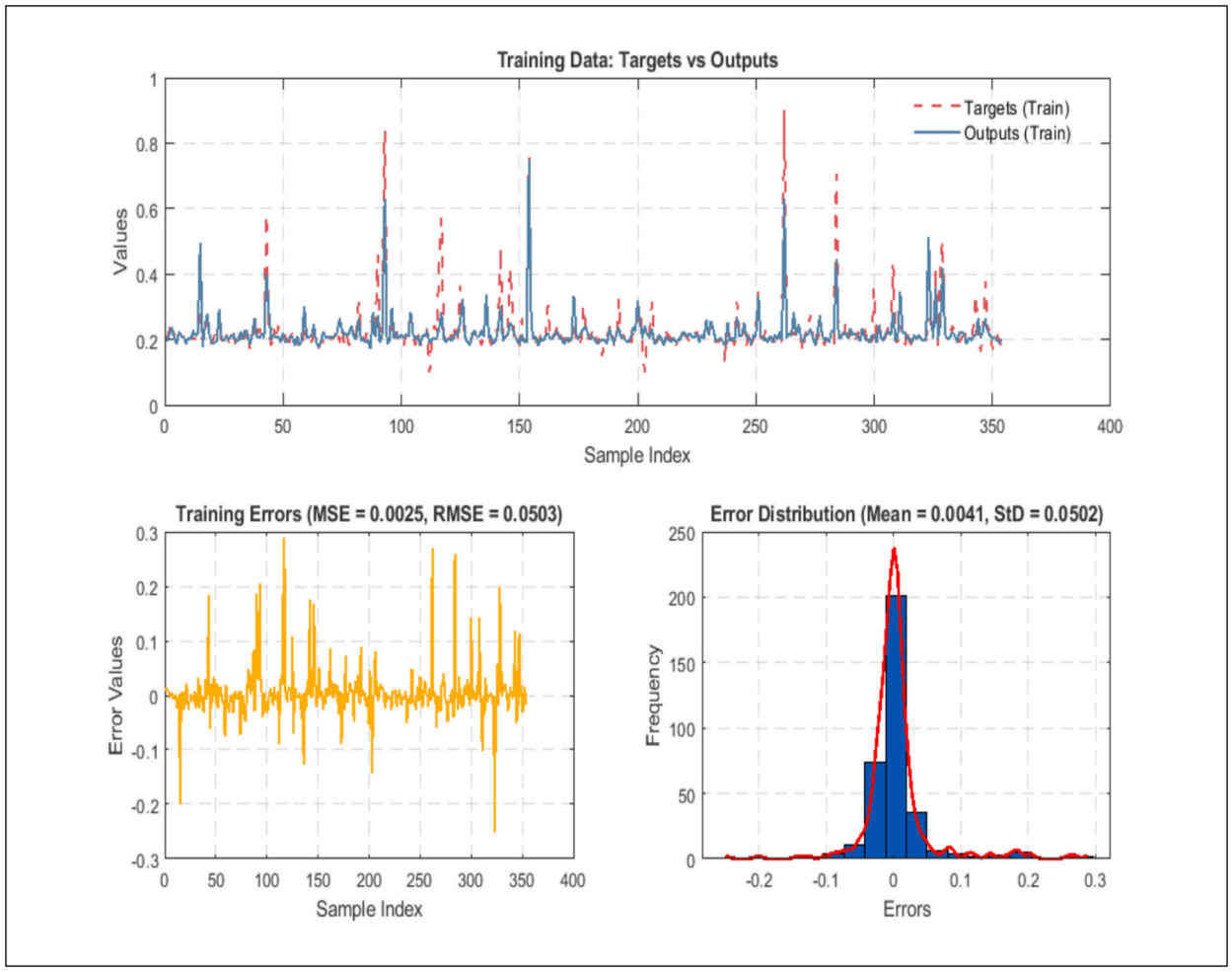

Fig. 7. Plots of WCA: actual vs predicted (top), training errors (bottom left) and error distribution (bottom right).

suggest the presence of noise or the need for further optimization to enhance generalization.

The results presented in Table 10 compare the performance of six optimization algorithms in terms of predictive accuracy (measured by MAE) and computational runtime. MFO has been selected due to its simple structure, very low number of control parameters, and exceptionally high citation count in the literature from 2017 to 2025, making it one of the most widely used benchmark algorithms in metaheuristic research; it continues to serve as a standard reference point even in recent studies. Similarly, WCA represents one of the most successful and robust algorithms of the earlier generation (2012–2015), particularly excelling in real-world engineering problems and constrained optimization tasks, where it consistently demonstrates a strong balance between exploration and exploitation and delivers competitive, reliable performance in practical applications. Alongside these two well-established reference methods, four more recent algorithms are evaluated: Harris Hawks Optimizer (HHO, 2019) [41], Slime Mould Algorithm (SMA, 2020) [42], Equilibrium Optimizer (EO, 2020) [43], and Archerfish Hunting Optimizer (AHO, 2022) [44]. This selection enables a meaningful assessment of whether state-of-the-art metaheuristic algorithms have achieved substantial improvements over powerful and still widely adopted classical approaches, both in terms of solution quality and computational efficiency.

The data presented in Table 10, which compares six optimization algorithms (MFO, WCA, HHO, AHO, SMA, EO) in terms of accuracy (via MAE) and runtime, reveals a classic trade-off between these two metrics. The WCA emerges as the most accurate method with the lowest error (MAE = 0.026006), but this superior precision comes at the cost of the longest computational time (2364.08 s). Conversely, the MFO algorithm is the fastest (1137.74 s) but exhibits a higher error (MAE = 0.030001). Algorithms such as HHO and AHO offer an attractive balance, maintaining accuracy close to WCA (with MAEs of 0.027500 and 0.028200, respectively) while requiring significantly less runtime (approximately 1800–1900 s). SMA demonstrates a reasonably balanced performance, achieving an MAE of 0.030800 comparable to MFO with a competitive runtime of 1680.40 s, positioning it as a viable option for problems requiring moderate accuracy and speed. In contrast, EO records the highest MAE (0.031500) with a runtime of 1750.60 s, making it the least competitive in this comparison. Therefore, the optimal algorithm choice is directly contingent on the problem's priorities: WCA for maximum accuracy where time is not a constraint, MFO for minimum runtime, and HHO or AHO for an optimal compromise between the two factors, while SMA offers a solid alternative when both accuracy and speed are moderately important.

Based on the detailed results presented in Table 11 which evaluates six metaheuristic algorithms (MFO, WCA, HHO, AHO, SMA, and EO) across six performance metrics on both training and test sets, a definitive performance ranking can be established. The WCA consistently ranks first, demonstrating the best overall accuracy and generalization. It achieves the lowest error metrics, with a training MSE of 0.002529 and a test MSE of 0.0027232, and the highest explanatory power, evidenced by a training R^2 of 0.64235 and a test R^2 of 0.40832. The HHO algorithm places a close second, with marginally higher errors (e.g., test MSE of 0.002850) and slightly lower R^2 values (test R^2 of 0.3750).

The MFO and AHO algorithms form a middle-performance group. For instance, MFO's test MAE (0.031123) and AHO's test R2 (0.3700) are

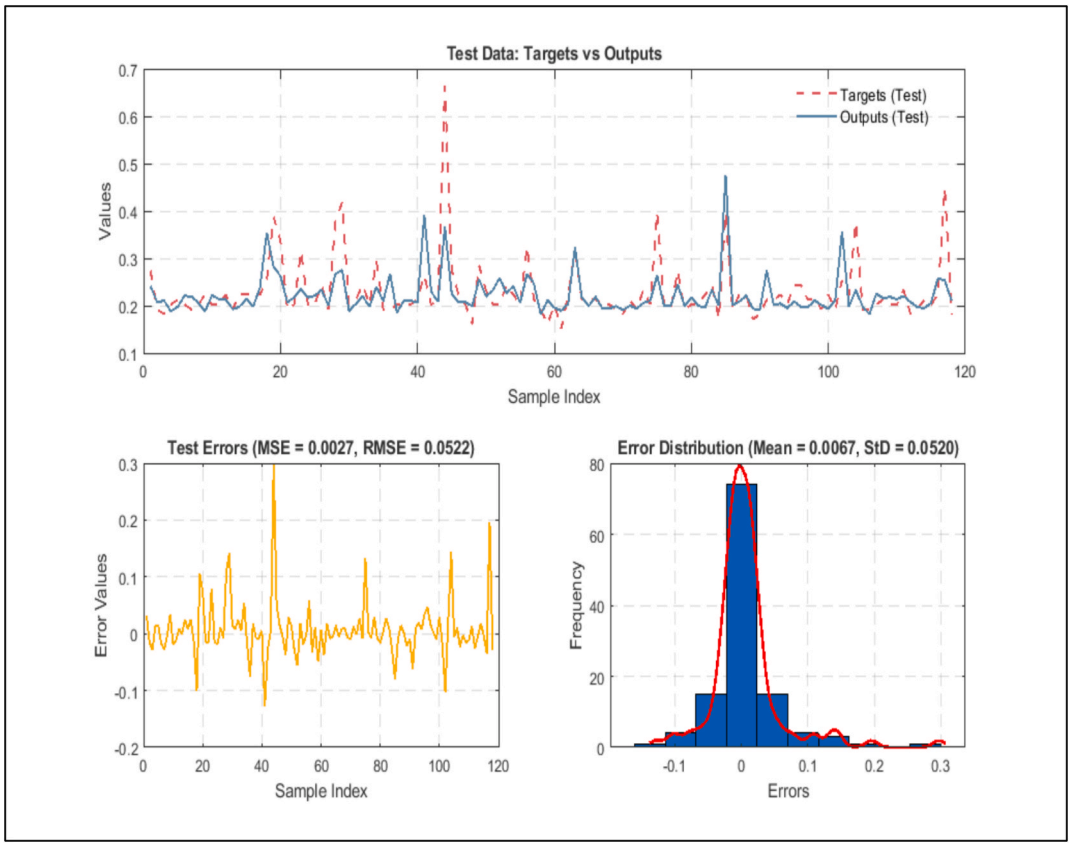

Fig. 8. Plots of WCA: actual vs predicted (top), testing errors (bottom left) and error distribution (bottom right).

Table 10
Optimal MAE values and optimization runtimes for MFO, WCA, and other metaheuristic algorithms.

Algorithm	Optimal value (MAE)	Optimization runtime (Seconds)
MFO	0.030001	1137.74
WCA	0.026006	2364.08
HHO	0.027500	1800.50
AHO	0.028200	1905.30
SMA	0.030800	1680.40
EO	0.031500	1750.60

noticeably inferior to WCA's results but remain superior to the lowest tier. In contrast, the SMA and EO consistently yield the weakest outcomes. EO, in particular, shows the poorest performance, registering the highest test MSE (0.003150), the lowest test R2 (0.3350), and the highest test MAPE (13.10%).

A critical trend observed across Table 11 is the significant performance decline from the training to the test phase for all algorithms. This is clearly illustrated by an average relative drop of approximately 39% in the NSE metric across all models, underscoring a common challenge with generalization. However, WCA manages this decline most effectively, with its NSE decreasing from 0.62974 to 0.3985 (a 36.7% drop), whereas algorithms like MFO experience a more severe reduction from 0.58021 to 0.35742 (a 38.4% drop), and HHO and AHO exhibit even larger declines of over 41%. This consistent numerical evidence solidifies WCA's position as the most robust and reliable algorithm for this predictive modeling task.

In the following analysis of the results, the prioritization and interpretation of the importance of input features by the best-performing model, namely the MLP optimized by WCA, are addressed. Feature importance indicates how critical an input feature is to the modeling process and helps identify which features are less effective or redundant. To this end, the permutation importance method [45,46], introduced by Breiman and Cutler [47,48], was employed in this study. In this method, after establishing a baseline accuracy by passing a test set through the model, a single column (i.e., a single feature) is permuted, and the test samples are passed through the model again to recalculate the accuracy. The permutation importance of the feature is determined by the difference between the baseline accuracy and the accuracy recalculated after permuting the column. In this study, MSE was used as the evaluation metric, and the permutation process was repeated 100 times. Table 12 presents the results of this method, which will be analyzed in detail in the following sections.

Table 12 ranks the variables based on their impact on the MLP optimized by WCA: AADT/Lane.mile, Link Length, Total rate, and PRCP/(1000). The top-ranked variable, AADT/Lane.mile (rank 1), which represents the average annual daily traffic per lane mile, highlights its critical role in travel time prediction. This suggests that traffic volume is a primary determining factor, as higher AADT values are typically associated with increased congestion and longer travel durations, making it essential for accurate forecasting in traffic management systems. Link Length (rank 2) is considered the second most influential factor, emphasizing the importance of road segment length in travel time estimation. Longer links naturally contribute to increased travel times, and this ranking underscores its significance in route planning and optimization. Total rate (rank 3) plays a moderate role, indicating that it indirectly affects travel time through operational conditions and is useful for identifying high-risk segments. Finally, PRCP/(1000) (rank 4), which represents precipitation adjusted by a factor, has the least impact, suggesting that weather conditions, while relevant, are less critical compared to traffic and structural factors in this context. This ranking implies that travel time prediction models should prioritize traffic volume and road length for the highest accuracy. These insights can guide real-time infrastructure planning.

Table 11
Training and testing performance metrics for MFO, WCA, and other metaheuristic algorithms.

Algorithm	Metric	Train Set	Test Set
MFO	MSE	0.0028673	0.0029091
	RMSE	0.053547	0.053936
	MAE	0.030001	0.031123
	R^2	0.5818	0.38401
	MAPE	12.01	12.078
	NSE	0.58021	0.35742
WCA	MSE	0.002529	0.0027232
	RMSE	0.050289	0.052184
	MAE	0.026006	0.03062
	R^2	0.64235	0.40832
	MAPE	10.15	11.476
	NSE	0.62974	0.3985
HHO	MSE	0.002650	0.002850
	RMSE	0.051480	0.053400
	MAE	0.027500	0.029800
	R^2	0.6200	0.3750
	MAPE	11.30	11.95
	NSE	0.6050	0.3550
AHO	MSE	0.002710	0.002910
	RMSE	0.052038	0.053936
	MAE	0.028200	0.030500
	R^2	0.6100	0.3700
	MAPE	11.50	12.05
	NSE	0.5980	0.3500
SMA	MSE	0.002920	0.003100
	RMSE	0.054070	0.055680
	MAE	0.030800	0.032900
	R^2	0.5600	0.3450
	MAPE	12.20	12.90
	NSE	0.5550	0.3400
EO	MSE	0.003000	0.003150
	RMSE	0.054770	0.056120
	MAE	0.031500	0.033400
	R^2	0.5400	0.3350
	MAPE	12.50	13.10
	NSE	0.5400	0.3300

Table 12
Permutation feature importance results for MLP optimized by WCA.

Feature	Rank
AADT/Lane.mile	1
Link Length	2
Total rate	3
PRCP/(1000)	4

6. Conclusions

This study presents a comprehensive evaluation of a combined MLP model with six well-known metaheuristic optimization algorithms (MFO, WCA, HHO, AHO, SMA, and EO) for travel time prediction, assessed across multiple performance metrics, including MAE, execution time, MSE, RMSE, R^2, MAPE, and NSE.

Among the evaluated algorithms, WCA consistently achieves the highest predictive accuracy, with the lowest error metrics (training MAE = 0.026006, test MAE = 0.03062; training MSE = 0.002529, test MSE = 0.0027232) and the strongest explanatory power (training R2 = 0.64235, test R2 = 0.40832). It should be noted that in this study, WCA's superiority is specifically based on accuracy metrics. Convergence analysis further highlights WCA's rapid initial decline in the objective function from 0.044 to 0.026 over 500 iterations surpassing MFO's reduction from 0.09 to 0.030, underscoring WCA's ability to reach a lower final objective value efficiently. While WCA excels in precision, MFO offers a faster alternative, illustrating that the ultimate choice between accuracy and computational efficiency depends on the specific requirements of the application.

HHO and AHO exhibit intermediate performance, achieving accuracy metrics close to WCA while requiring moderately less runtime, whereas SMA and EO consistently produce the weakest predictive results. Across all performance metrics, the ranking of algorithms demonstrates a clear trend: WCA provides the highest accuracy, MFO balances speed and accuracy, and the other algorithms achieve intermediate or lower performance levels.

Overall, these results highlight a trade-off between predictive accuracy and computational efficiency, emphasizing that algorithm selection should be guided by the priorities of the application. For scenarios where maximum accuracy is the primary objective, WCA is recommended, whereas MFO or HHO/AHO may be preferable in cases where computational cost or runtime is a critical factor.

7. Applications in transportation management and ethical considerations

The optimized ANN models using MFO and WCA offer significant potential for enhancing transportation management; however, their deployment in public transportation planning raises critical ethical considerations. Transparency is paramount, as stakeholders must understand how these data-driven models generate predictions for real-time TTI systems or identify high-crash-rate zones, ensuring public trust in decision-making processes. Accountability is equally crucial, as errors in model outputs, such as inaccurate travel time estimates or misidentified safety risks, could lead to inefficient resource allocation or compromised road safety. Moreover, data biases pose a substantial risk, particularly when training datasets may underrepresent certain regions, demographics, or environmental conditions (e.g., Precipitation or Total Crash Rate variations). Such biases could skew model predictions, disproportionately affecting underserved communities or leading to inequitable urban planning outcomes. To mitigate these concerns, future implementations should prioritize explainable AI frameworks, rigorous bias audits, and inclusive data collection strategies to ensure equitable and responsible use of these models in ITS and urban planning.

8. Limitations

This study has several limitations that should be acknowledged when interpreting the results. One of the primary limitations is the observed performance degradation from the training dataset to the test dataset, which indicates the presence of overfitting. This behavior is commonly observed in neural network–based models, particularly when applied to relatively small datasets with limited variability. The dataset used in this study consists of 472 observations with four explanatory variables, which may constrain the model's ability to fully capture complex traffic dynamics and generalize to unseen real-world conditions. While the adopted train–test split provides a reasonable balance between model learning and validation, the limited data volume inherently restricts the robustness of neural network training. In addition, the analysis was conducted using fixed hyperparameter settings, including population size, number of iterations, and network architecture. Although these parameters were selected based on prior studies and trial-and-error experimentation, alternative configurations or more extensive hyperparameter tuning may further influence model performance and generalization. Furthermore, the study relies on annual average data, which limits its applicability to short-term or real-time travel time reliability prediction. Temporal variations such as hourly traffic fluctuations, incident severity, and dynamic weather patterns were not

explicitly modeled, which may affect predictive accuracy in operational contexts. These limitations highlight the need for larger and more diverse datasets, additional explanatory variables, and more advanced validation strategies in future research to improve model robustness and generalization capability.

9. Future studies

Future research could explore several promising avenues to enhance the performance and applicability of metaheuristic and machine learning approaches. One potential direction is investigating hybrid methods that combine the strengths of MFO and WCA alongside other metaheuristic techniques. Additionally, a more systematic and extensive optimization of neural network architectures including the number of neurons, number of layers, activation functions, and a broader set of hyperparameters using multiple optimization strategies, such as grid search, random search, or metaheuristic-based optimization, could further improve predictive accuracy and computational efficiency. Sensitivity analyses on key parameters such as initial population size, learning rate, and other critical hyperparameters, in conjunction with architectural optimization, can provide deeper insights into model performance. Incorporating advanced machine learning techniques, such as LSTM, CNN, and Conv-LSTM, as well as comparing multiple ANN architectures, can provide a comprehensive perspective on scalability and efficiency.

The current methodology demonstrates good potential for scalability to larger datasets, particularly with the increasing availability of high-resolution probe vehicle data and connected vehicle information. Although metaheuristic optimization becomes more computationally demanding with larger sample sizes, the population-based nature of MFO and WCA allows straightforward parallelization. Moreover, hybrid strategies (metaheuristic initialization followed by gradient-based fine-tuning) could be explored to improve efficiency on very large datasets.

Regarding geographic applicability, while the model was trained on Virginia interstate data, the selected input features are widely collected by transportation agencies in many countries. The dominance of traffic volume and segment length as key predictors suggests that the approach should be reasonably transferable to other regions, although local calibration (re-optimization or transfer learning) would likely be necessary to account for differences in driving culture, road typology, climate, and incident patterns.

Furthermore, given that the current study used a dataset with only 472 observations and four features, testing these models on larger and more diverse datasets, multi-regional datasets, or integrating real-time streaming data is necessary to better assess their robustness under varying conditions, such as traffic fluctuations or different environmental contexts.

Multi-regional validation and investigation of domain adaptation methods are recommended in future work to enhance broader applicability and practical relevance of the proposed framework [49,50,51].

To ensure reliable results with metaheuristic algorithms, which are inherently variable due to their random and iterative nature, multiple independent runs typically 30 to 50, depending on problem complexity are recommended, provided sufficient computational resources are available. This approach allows capturing performance variability through metrics like convergence rates or objective function values. Statistical tests, such as t-tests, ANOVA, or the Wilcoxon rank-sum test, should then be applied to determine whether observed performance differences are statistically significant (e.g., at a p-value of 0.05) rather than random. Additionally, calculating mean, median, standard deviation, and confidence intervals further supports a robust evaluation of algorithm stability and reliability, ensuring that conclusions regarding methods like MFO or WCA are evidence-based.

10. Ethical approval and consent to participate

Authors declare that this paper adheres to ethical standards by ensuring confidentiality, promoting transparency in data collection and analysis, and upholding integrity and honesty in reporting and acknowledging all sources of information.

11. Consent for publication

Not applicable, as this study does not involve human participants or identifiable data requiring consent for publication.

12. Availability of supporting data

The datasets generated and/or analyzed during this study are available from the corresponding author upon reasonable request.

13. Authors' contributions

All authors have contributed equally to the study.

CRediT authorship contribution statement

Navid Khorshidi: Writing – original draft, Visualization, Validation, Resources, Data curation, Conceptualization. **Soheil Rezashoar:** Writing – original draft, Visualization, Validation, Software, Formal analysis. **Shahriar Afandizadeh Zargari:** Supervision, Project administration. **Hamid Mirzahossein:** Writing – review & editing, Supervision, Project administration, Methodology, Conceptualization.

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this study.

Declaration of competing interest

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

Acknowledgments

There is no Acknowledgment part for this paper.

Data availability

Data will be made available on request.

References

[1] C. Systematics, N.R. Council, *Analytical procedures for determining the impacts of reliability mitigation strategies*, Transportation Research Board, 2013.
[2] R.G. Dowling K.L. Parks B. Nevers J. Josselyn S. Gayle, Incorporating travel-time reliability into the congestion management process: a primer, Retrieved from 2015, http://www.ops.fhwa.dot.gov/publications/fhwahop14034/fhwahop14034.pdf.
[3] K. Lyman, R.L. Bertini, Using travel time reliability measures to improve regional transportation planning and operations, Transp. Res. Rec. 2046 (1) (2008) 1–10, https://doi.org/10.3141/2046-01.
[4] C. Systematics, Texas Transportation Institute. *Traffic Congestion and Reliability: Trends and Advanced Strategies for Congestion Mitigation,* 2005. Retrieved from https://rosap.ntl.bts.gov/view/dot/20656/dot_20656_DS1.pdf.
[5] T. Shaw, Performance Measures of Operational Effectiveness for Highway Segments and Systems, Transportation Research Board, vol. 311, 2003.
[6] Goodwin, P. (2004). The economic costs of road traffic congestion. Retrieved from https://discovery.ucl.ac.uk/id/eprint/1259/1/2004_25.pdf.
[7] D. Schrank, T. Lomax, The 2003 urban mobility report, 2003. Retrieved from https://rosap.ntl.bts.gov/view/dot/15904/dot_15904_DS1.pdf.
[8] B. Kuhn L. Higgins A. Nelson M. Finley G. Ullman S. Chrysler K. Wunderlich V. Shah C. Dudek Lexicon for conveying travel time reliability information

(0309273072), 2014 Retrieved from https://www.nationalacademies.org/read/22604/chapter/1.
[9] Z. Chen, W. Fan, Data analytics approach for travel time reliability pattern analysis and prediction, J. Modern Transport. 27 (4) (2019) 250–265, https://doi.org/10.1007/s40534-019-00195-6.
[10] W. Pu, Analytic relationships between travel time reliability measures, Transp. Res. Rec. 2254 (1) (2011) 122–130, https://doi.org/10.3141/2254-13.
[11] D.P. Gaver Jr, Headstart strategies for combating congestion, Transp. Sci. 2 (2) (1968) 172–181, https://doi.org/10.1287/trsc.2.2.172.
[12] A. Richardson, M. Taylor, Travel time variability on commuter journeys, Retrieved from, High Speed Ground Transport. J. 12 (1) (1978), https://trid.trb.org/View/80100.
[13] J.N. Prashker, Direct analysis of the perceived importance of attributes of reliability of travel modes in urban travel, Transportation 8 (4) (1979) 329–346, https://doi.org/10.1007/bf00167987.
[14] L.A. Senna, The influence of travel time variability on the value of time, Transportation 21 (1994) 203–228, https://doi.org/10.1007/BF01098793.
[15] L. Elefteriadou, X. Cui, *A framework for defining and estimating travel time reliability*, Retrieved from (2007). https://trid.trb.org/View/801806.
[16] Y. Zhang, Y. Liu, Analysis of peak and non-peak traffic forecasts using combined models, J. Adv. Transp. 45 (1) (2011) 21–37, https://doi.org/10.1002/atr.128.
[17] CP, K. Karuppanagounder, Performance prediction model for urban dual carriageway using travel time-based indices, Transp. Dev. Econ. 6 (1) (2020) 2, https://doi.org/10.1007/s40890-019-0090-8.
[18] S. Afandizadeh Zargari, N. Amoei Khorshidi, H. Mirzahossein, N. Kalantari, Comparative approach for predicting travel time reliability (a case study of Virginia interstate), Innovative Infrastructure Solutions 6 (4) (2021) 229, https://doi.org/10.1007/s41062-021-00597-8.
[19] D. Billings, J.-S. Yang, *Application of the ARIMA models to urban roadway travel time prediction-a case study*. Paper Presented at the 2006 IEEE International Conference on Systems, Man and Cybernetics, 2006.
[20] V. Belle, I. Papantonis, Principles and practice of explainable machine learning, Front. Big Data 4 (2021) 688969, https://doi.org/10.3389/fdata.2021.688969.
[21] J.S. Almeida, Predictive non-linear modeling of complex data by artificial neural networks, Curr. Opin. Biotechnol. 13 (1) (2002) 72–76, https://doi.org/10.1016/s0958-1669(02)00288-4.
[22] E.W.M. Lee, C.P. Lim, R.K.K. Yuen, S. Lo, A hybrid neural network model for noisy data regression, IEEE Trans. Syst. Man Cybernet. Part B (Cybernet.) 34 (2) (2004) 951–960, https://doi.org/10.1109/TSMCB.2003.818440.
[23] W. Pannakkong, K. Thiwa-Anont, K. Singthong, P. Parthanadee, J. Buddhakulsomsiri, Hyperparameter tuning of machine learning algorithms using response surface methodology: a case study of ANN, SVM, and DBN, Math. Probl. Eng. 2022 (1) (2022) 8513719, https://doi.org/10.1155/2022/8513719.
[24] N. Khorshidi, S. Rezashoar, P. Amini, S.A. Zargari, H.J.T.E. Mirzahossein, Metaheuristic-Optimized Neural Networks for Travel Time Index Prediction: a Comparative Study of Wild Horse and Coot Optimization Algorithms. 100395 (2025), https://doi.org/10.1016/j.treng.2025.100395.
[25] N. Khorshidi, S.A. Zargari, S. Rezashoar, H.J.M.L.W.A. Mirzahossein, Optimizing Travel Time Reliability with XAI: A Virginia Interstate Network Case Using Machine Learning and Meta-Heuristics (2025), https://doi.org/10.1016/j.mlwa.2025.100709.
[26] G. Mustafa, M.S. Khattak, S. Ishfaq, M.T. Afzal, Q. Mahmood, Y.N.J.E.I. Khalid, Interpretable Genetic Programming with SHAP-Guided Multi-Objective Optimization for Scientific Impact Modeling. 19 (1) (2026) 14, https://doi.org/10.1007/s12065-025-01125-8.
[27] M.-C. Popescu, V.E. Balas, L. Perescu-Popescu, N. Mastorakis, Multilayer perceptron and neural networks, WSEAS Trans. Circ. Syst. 8 (7) (2009) 579–588, https://doi.org/10.5555/1639537.1639542.
[28] O.R. Adegboye, E.D. Ülker, A.K. Feda, E.B. Agyekum, W.F. Mbasso, S.J.H. Kamel, Enhanced multi-layer perceptron for CO2 emission prediction with worst moth disrupted moth fly optimization, WMFO 10 (11) (2024), https://doi.org/10.1016/j.heliyon.2024.e31850.
[29] H. Mohan, G. Agrawal, V. Jately, A. Sharma, B.J.M. Azzopardi, Neural Network-Driven Sensorless Speed Control of EV Drive Using PMSM. 11 (19) (2023) 4029, https://doi.org/10.3390/math11194029.
[30] L.B. Almeida, Multilayer perceptrons, in: *Handbook of Neural Computation*, CRC Press, 2020, pp. C1. 2: 1-C1. 2: 30.
[31] S. Mirjalili, Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm, Knowl.-Based Syst. 89 (2015) 228–249, https://doi.org/10.1016/j.knosys.2015.07.006.
[32] Z.J.A.I. Yang, FMFO: Floating flame moth-flame optimization algorithm for training multi-layer perceptron classifier *53*(1) (2023) 251–271. doi: 10.1007/s10489-022-03484-6.
[33] B. Das S. Roy N. Debbarma P.J.N.C. Bhattacharya, Applications, A comparative study of hybrid neural network with metaheuristic algorithm for breast cancer data classification with TOPSIS MCDM approach, 2025, pp. 1–26. doi: 10.1007/s00521-025-11281-8.
[34] N.N. Khan Y. Mahale K. Kulkarni S. Pant A. Kumar K.J.N.-I.O.A.F. C.-P. S. Kotecha, An Overview of Nature-Inspired Optimization Techniques for Smart Cities, 2025. pp. 193–250. doi: 10.4018/979-8-3693-6834-3.ch007.
[35] H. Eskandar, A. Sadollah, A. Bahreininejad, M. Hamdi, Water cycle algorithm–a novel metaheuristic optimization method for solving constrained engineering optimization problems, Comput. Struct. 110 (2012) 151–166, https://doi.org/10.1016/j.compstruc.2012.07.010.
[36] A. Sadollah, H. Eskandar, H.M. Lee, D.G. Yoo, J.H. Kim, Water cycle algorithm: a detailed standard code, SoftwareX 5 (2016) 37–43, https://doi.org/10.1016/j.softx.2016.03.001.
[37] K. Jahan, S.M. Pradhanang, Predicting runoff chloride concentrations in suburban watersheds using an artificial neural network (ANN), Hydrology 7 (4) (2020) 80, https://doi.org/10.3390/hydrology7040080.
[38] M.A. Awadallah I. Abu-Doush M.A. Al-Betar M.S. Braik, Metaheuristics for optimizing weights in neural networks, in: *Comprehensive Metaheuristics*, Elsevier, 2023, pp. 359–377.
[39] S. Park, S. Son, J. Bae, D. Lee, J.-J. Kim, J. Kim, Robust spatiotemporal estimation of PM concentrations using boosting-based ensemble models, Sustainability 13 (24) (2021) 13782, https://doi.org/10.3390/su132413782.
[40] A. Ashrafian, M.J. Taheri Amiri, P. Masoumi, M. Asadi-shiadeh, M. Yaghoubi-chenari, A. Mosavi, N. Nabipour, Classification-based regression models for prediction of the mechanical properties of roller-compacted concrete pavement, Appl. Sci. 10 (11) (2020) 3707, https://doi.org/10.3390/app10113707.
[41] A.A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: algorithm and applications, Futur. Gener. Comput. Syst. 97 (2019) 849–872, https://doi.org/10.1016/j.future.2019.02.028.
[42] S. Li, H. Chen, M. Wang, A.A. Heidari, S. Mirjalili, Slime mould algorithm: a new method for stochastic optimization, Futur. Gener. Comput. Syst. 111 (2020) 300–323, https://doi.org/10.1016/j.future.2020.03.055.
[43] A. Faramarzi, M. Heidarinejad, B. Stephens, S. Mirjalili, Equilibrium optimizer: a novel optimization algorithm, Knowl.-Based Syst. 191 (2020) 105190, https://doi.org/10.1016/j.knosys.2019.105190.
[44] F. Zitouni, S. Harous, A. Belkeram, L.E.B. Hammou, The archerfish hunting optimizer: a novel metaheuristic algorithm for global optimization, Arab. J. Sci. Eng. 47 (2) (2022) 2513–2553, https://doi.org/10.1007/s13369-021-06208-z.
[45] A. Altmann, L. Toloşi, O. Sander, T. Lengauer, Permutation importance: a corrected feature importance measure, Bioinformatics 26 (10) (2010) 1340–1347, https://doi.org/10.1093/bioinformatics/btq134.
[46] A. Fisher, C. Rudin, F. Dominici, All models are wrong, but many are useful: learning a variable's importance by studying an entire class of prediction models simultaneously, J. Mach. Learn. Res. 20 (177) (2019) 1–81, https://doi.org/10.48550/arXiv.1801.01489.
[47] L. Breiman, Random forests, Mach. Learn. 45 (1) (2001) 5–32.
[48] D.R. Cutler, T.C. Edwards Jr, K.H. Beard, A. Cutler, K.T. Hess, J. Gibson, J. J. Lawler, Random forests for classification in ecology, Ecology 88 (11) (2007) 2783–2792, https://doi.org/10.1890/07-0539.1.
[49] G. Mustafa, M.T. Afzal, A. Rauf, M.A.J.E.I. Khan, Beyond Publication Numbers: a Novel Approach to Academic Ranking Using Evolutionary Programming. 18 (3) (2025) 62, https://doi.org/10.1007/s12065-025-01046-6.
[50] G. Mustafa A. Rauf M.T.J.I.J. Afzal, O.D.S. Analytics, Enhancing Author Assessment: an Advanced Modified Recursive Elimination Technique (MRET) for Ranking Key Parameters and Conducting Statistical Analysis of Top-Ranked Parameter 20(2) (2025) 1461–1483. doi: 10.1007/s41060-024-00545-6.
[51] G. Mustafa A. Rauf A.S. Al-Shamayleh M. Sulaiman W. Alrawagfeh M.T. Afzal A.J.I. A. Akhunzada, Optimizing Document Classification: Unleashing the Power of Genetic Algorithms 11 (2023) 83136–83149. doi: 10.1109/ACCESS.2023.3292248.
[52] J. Van Lint, H.J. Van Zuylen, Monitoring and predicting freeway travel time reliability: using width and skew of day-to-day travel time distribution, Transp. Res. Rec. 1917 (1) (2005) 54–62, https://doi.org/10.1177/0361981051917001007.
[53] Z. Wang, A. Goodchild, E. McCormack, A methodology for forecasting freeway travel time reliability using GPS data, Transp. Res. Procedia 25 (2017) 842–852, https://doi.org/10.1016/j.trpro.2017.05.461.
[54] X. Zhang, M. Chen, Quantifying the impact of weather events on travel time and reliability, J. Adv. Transp. 2019 (1) (2019) 8203081, https://doi.org/10.1155/2019/8203081.
[55] X.C. Liu N. Haghighi, Travel-Time Reliability in simulation and Planning Models: Utah Case Study (SHRP2 L04 IAP Round 7), 2019. Retrieved from https://rosap.ntl.bts.gov/view/dot/54020.
[56] R. Amrutsamanvar, G. Joshi, S.S. Arkatkar, R.S. Chalumuri, Em*pirical travel time reliability assessment of indian urban roads,* Paper presented at the Recent Advances in Traffic Engineering: Select Proceedings of RATE 2018, 2020.
[57] Z. Wu, L.R. Rilett, W. Ren, New methodologies for predicting corridor travel time mean and reliability, Int. J. Urban Sci. 26 (3) (2022) 517–540, https://doi.org/10.1080/12265934.2021.1899844.
[58] S.A. Zargari, N.A. Khorshidi, H. Mirzahossein, H. Heidari, Analyzing the effects of congestion on planning time index–Grey models vs. random forest regression. Int. J. Transport. Sci. Technol. 12(2) (2023) 578-593. doi: 10.1016/j.ijtst.2022.05.008.
[59] M. Zhao, X. Zhang, J. Appiah, M.D. Fontaine, Travel time reliability prediction using random forests, Transp. Res. Rec. 2678 (3) (2024) 531–545, https://doi.org/10.1177/0361198123118214.
[60] S. Afandizadeh, N. Amoei Khorshidi, H. Mirzahossein, S. Shakoori, Predicting the fluctuation of travel time reliability as a result of congestion variations by bagging-based regressors, Civil Eng. Infrastruct. J. 57 (1) (2024) 85–101, https://doi.org/10.22059/ceij.2023.349853.1878.
[61] B. Anil Kumar, G. Chandana, L. Vanajakshi, Travel time reliability prediction using quantile random forest regression, Transp. Dev. Econ. 11 (1) (2025) 19, https://doi.org/10.1007/s40890-025-00239-z.
[62] N. Khorshidi, S.A. Zargari, H. Mirzahossein, H. Heidari, T. Waller, Predicting travel-time reliability in road networks: a fitrnet-based approach–a case study of england, Sci. J. Siles. Univ. Technol. Ser. Transport 126 (2025) 79–95, https://doi.org/10.20858/sjsutst.2025.126.5.