



IJRASET

International Journal For Research in
Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: <https://doi.org/10.22214/ijraset.2026.81829>

www.ijraset.com

Call:  08813907089

E-mail ID: ijraset@gmail.com

Real Time Traffic Condition Based Smart Signal System

Gavini Venkateswari, Yetra Pujitha Gayathri, Meka Naga Phani, Nandam Hurmi Nandalal
Bapatla Women's Engineering College, India

Abstract: In the era of crowd delivery vehicle routing and traffic management in smart cities, a complex challenge appears indistinctly, affecting both developed and developing nations worldwide. This challenging problem involves optimizing multi-depot routes while addressing various hurdles: minimizing travel time, distance, fuel consumption, and carbon emissions, all while navigating dynamic traffic congestion across diverse pathways. Existing approaches often focus on isolated aspects like shortest paths, carbon emissions, or traffic prediction, leaving the comprehensive multi-depot traffic management problem unaddressed. In response, this research work proposes an Intelligent Multi-Depot Vehicle Routing and Management (IM-VRM) model which provides a comprehensive and holistic solution. It employs a Graph Neural Network (GNN) learning-based routing with a greedy optimization to establish initial optimal pathways for multi-depot journeys. Subsequently, the IM-VRM model integrates traffic congestion prediction with green parameter computation, engaging the Dijkstra algorithm to select the most admissible routes. This consecutive steps-based travel route guidance process optimizes routing for heterogeneous vehicles, including both heavy-duty and light-duty types. It accounts for load-dependent fuel consumption, velocity, and carbon emissions. By doing so, it simplifies the complexities of multi-depot traffic routing and management. The proposed model has been rigorously evaluated using a real-world multi-depot traffic dataset, demonstrating its practical viability. Notably, IM-VRM model achieves a remarkable improvement in fuel savings, reduced carbon emissions, and shorter travel time outperforming previous state-of-the-art methods in both efficiency and precision.

Index Terms— Crowd delivery, carbon emission, congestion, multi-depot rides, traffic management.

I. INTRODUCTION

IN MODERN urban landscapes, the efficient coordination of delivery routes across multiple depots and vehicle traffic management in the smart cities presents significant challenges [1],[2]. The increasing number of delivery vehicles intensifies urban congestion, leading to higher fuel consumption [3], air pollution, and carbon dioxide (CO_2) emissions [4],[5]. While strategies such as car-pooling and crowd delivery have been proposed, they often suffer from inadequate implementation and management. Post-COVID trends have further exacerbated the situation, with emissions from individual activities, such as taxi rides and delivery services, having risen up noticeably. Delivery vehicles, particularly those used for food delivery, are major contributors to air pollution, with the transportation sector responsible for approximately 28% of global greenhouse gas emissions [6],[7]. The growing prevalence of online food delivery services has led to a 20% increase in Vehicle Miles Traveled (VMT) in the United States from 2010 to 2020 [8], further straining urban infrastructure. This surge in delivery vehicles exacerbates traffic congestion and results in excessive emissions of pollutantssuch as CO_2 , nitrogen oxides, and particulate matter [9],[10],[11].

To address these challenges, there is a critical need for scalable, sustainable, and computationally efficient solutions that can meet the growing demands of urban logistics while mitigating the environmental impact of delivery systems in smart city environments. The critical question is how to develop an effective framework for optimizing multi-depot delivery operations in real-time that minimizes travel times, distances, fuel consumption, and emissions while dynamically adapting to fluctuating traffic conditions.

A. Related Work

Several innovative vehicle routing strategies have been proposed for efficient delivery management in smart cities. Jolfaei and Alinaghian [12] addressed the multi-depot vehicle routing problem with roaming delivery locations and hard time windows, presenting a mathematical model and two heuristic algorithms, including a hybrid evolutionary logical search-based approach for large-scale solutions. CrowdNet, a spatial crowdsourcing-based system was proposed in [13], leveraging existing taxis for food delivery, using evolutionary algorithms, improved insertion approaches, and a ranking module based on modern portfolio theory to optimize delivery cost and timeliness. Chen [14] and Kouziokas et al.

[15] proposed NN-based Deep Reinforcement Learning based routing and search algorithms to predict traffic congestion. Joe and Lau [16] proposed the Deep Reinforcement Learning Approach for Dynamic Vehicle Routing Problems, combining neural network-based learning with stochastic routing heuristics. Wang and Su [17] proposed a two-stage control strategy for CAV platoons, focusing on fast platoon formation and fuel-efficient, safe junction crossings, optimized for urban traffic management. Liu et al. [18] suggested a Green Vehicle Routing Optimization algorithm for emission reduction. Abbatecola et al. [19] proposed a Decision Support Approach for postal delivery and waste collection, optimizing vehicle assignment and routing. Perboli et al. [20] presented a multi-stage heuristic based on simulation optimization for dynamic and stochastic vehicle routing problems. Chen et al. [21] proposed CROWDDELIVER, utilizing occupied cabs for city-wide package delivery while Du et al. [13] presented CrowdNet, a system leveraging spatial crowdsourcing for food delivery. Kaffle et al. [22] designed a crowd-source-enabled system for last-leg deliveries, reducing overall transportation costs. Hong et al. [23] presented a framework for improved crowd-sourcing multi-hop package delivery, utilizing spare capacity in urban vehicles.

B. Research Gap and Paper Contributions

Despite numerous proposed strategies for managing delivery time, CO₂ emission, and travel distance constrained with mitigation of dynamic traffic congestion, types, etc. [12], [13], [14], [15], [16], [17], [18], [19], [20], [21], [24]. There remains a need for a comprehensive and ride-practical solution that is highly effective and less complex ensuring seamless integration into the ever-evolving smart city landscape. In this context, this paper proposed an **Intelligent Multi-Depot Vehicle Routing and Management (IM-VRM)** model which provides a comprehensive and holistic solution. The key contributions are fourfold:

- A novel multi-depot crowd delivery vehicle routing management model is proposed which addresses multiple challenges including optimized fuel consumption, travel time, CO₂ emission, and travel distance constrained with mitigation of dynamic traffic congestion.
- An innovative concept of employing Greedy optimization within a GNN is introduced to determine optimal route selection for multi-depot rides.
- IM-VRM model illustrates seamless integration of traffic congestion prediction with the computation of green parameters to determine the most optimal routes. The progressive, step-based travel route guidance process not only streamlines the intricate challenges of multi-depot traffic routing and management but also surpasses the complexity of existing state-of-the-art methods.
- Experimental simulations and evaluations using real-world benchmark data demonstrate that the IM-VRM model outperforms conventional and state-of-the-art (SOTA) approaches across key performance metrics for dynamic transport management.

C. Paper Organization

The considered problem is formulated in Section II “Problem Formulation” which describes the challenges, assumptions, and constraints undertaken associated with the addressed multi-depot crowd delivery routing management problem. The proposed method is discussed in Section III “IM-VRM Model” which includes travel route network optimization using distance-oriented greedy optimization of travel route GNN (in Sub-section III-A) followed by traffic congestion handling with green parameters optimization (Sub-section III-B); and concise discussion of operational design and complexity computation (Sub-section IV). Section V “Performance Evaluation” discusses experimental setup, implementation, dataset description, and results of initial route selection, handling traffic congestion, green parameters constrained optimal delivery route selection, and comprehensive performance analysis of the proposed model versus without proposed model. Section VI “Conclusion” entails conclusive remarks and the future scope of the proposed work. The symbols along with their meaning used throughout the paper are described in Table I.

II. PROBLEM FORMULATION

Consider n single-depot or multi-depot delivery rides requests $\{R_1, R_2, \dots, R_n\}_{R_1}$ for a commencing m or n starting m or n possible

travel routes such as $\{S:D, D, \dots, D_a\}, \{S:D,$

systems in smart cities, they often face challenges in real-

world scalability, adaptability with dynamically varying traffic

$D_2^2, \dots, D_b^2\}$, and $\{S^n: D_1^n, D_2^n, \dots, D_c^n\}$, wherein,

R R 1R 2R R 1

conditions, congestion, and integration of diverse vehicle

$a_{th} \geq 1, b \geq 1, \text{ and } R_i c \geq 1; S_{th}^{Ri}$ represents source site of

i ride (R_i) and D specifies j

destination site of i

An Intelligent Traffic Organizer and Management Engine (ITOME) receives route selection requests and generates n responses respective to each request. ITOME operates the city traffic by using three knowledge databases including Live City Traffic (kd-LCT) which are maintained and updated with the help of Global Positioning System (GPS), capturing the live traffic movement; Vehicle Routes Connecting City (kd-VRCC) containing all travel routes connecting every site within the city; and Historical Traffic Data Repository (kd-HTDR) which keeps track records of previous temporal vehicle traffic about

TABLE I
LIST OF SYMBOLS AND NOTATIONS

Symbol	Meaning	Symbol	Meaning
R	Ride request	N	Maximum load of vehicle
S	Source site	C	Fuel cost
D	Destination site	E_k	Ending time
t	Time instance	B_k	Beginning time
OR	Optimal route	V	Set of nodes
V_h	Heterogeneous vehicle	VH	Set of vehicles
τ	Traffic condition	A	Set of routes
TW	Time Window	U	Set of vehicle types
AP	Air pollution	M	Set of all time periods
TD	Travel distance	E	Set of edges
TC	Travel cost	F_u	Fuel tank capacity
V_h^{HD}	Heavy duty vehicle	Θ^{status}	Traffic congestion status
V_h^{LD}	Light duty vehicle	φ	Latitude
S	Speed	Ω	Longitude
L	Travel load	v	Node in GNN
G	GNN	θ	Central angle
ds	Historical traffic data	D^{Ac}	Actual data
BL	Base learner	D^{Pr}	Predicted data
F	Feature vector	O	Predicted output by BL
X_{ij}	Route ($i; j$) traveled status	u	vehicle type

each route of the city. The core objective of ITOME is to map every online (i.e., currently moving vehicle) or offline (i.e., standby vehicle) requested ride with the most admissible routes considering the situation of live city traffic for t^{th} time instance. The essential challenges with intended constraints and design goals with assumptions are outlined in the subsequent sub-sections.

A. Challenges and Assumptions

ITOME possesses the request data $\{R_1, R_2, \dots, R_n\}$ that need to be addressed with the most admissible route selection proactively while addressing the following critical challenges:

- **Heterogeneous vehicles** $\{V_h1, V_h2, \dots, V_hn\} \in VH$ with different capacities, constraints, and capabilities impose vehicle specific constraints to be considered during optimal route $\{OR_1, OR_2, \dots, OR_n\}$ selection.
- The ever-changing traffic conditions $\{\tau_1, \tau_2, \dots, \tau_{cr}\}$ on city routes present a **dynamic adaptability challenge** for generating real-time or near-real-time solutions.
- As the number of depots and vehicles increases, the computational complexity of generating optimal delivery routes grows exponentially. This poses a critical challenge to the capability of finding optimal solutions $\{OR_1, OR_2, \dots, OR_n\}$ within a reasonable timeframe.
- Incorporating **time windows (TW), air pollution (AP), travel distance (TD), and travel cost (TC)** into the routing problem adds complexity and requires adherence to customer service time constraints.

Aligning with the above challenges, the following assumptions are considered:

- The heterogeneity of vehicle is restricted to heavy-duty vehicle type (V_h^{HD}) and light-duty vehicle type (V_h^{LD}) such that $\{H, D, L, D\} \in U$.
- The delivery rides $\{R_1, R_2, \dots, R_n\}$ apply specific time windows: $\{TW_1, TW_2, \dots, TW_n\}$ which is strictly incorporated into the routing problem.
- IM-VRM model decides the priority of $TW, AP, TD,$ and TC while requesting for the optimal route.

- ITOME is linked with kd-LCT, kd-VRCC, and kd-HTDR of each travel route connecting the city site.

B. Design Goals

Based on the problem description, intended challenges, and assumptions, the design goals for ITOME are as follows:

1) *Vehicle Fuel Consumption*: A vehicle fuel consumption model is developed using [25] which is derived from the Comprehensive Module Emissions Model (CMEM) by [26] for estimation of vehicles' fuel consumption rates ($FCV_{hu}(S,L)$) as a function of vehicle speed S and load L . Considering a heavy-duty vehicle (Vh^{HD}) traveling on a road with no slope, the VFCR function is stated in Eq. (1);

$$FCV_{hu}(S,L) = \alpha_1 V_{hu} S^{-1} + \alpha_2 V_{hu} S^2 + \alpha_3 V_{hu} + \alpha_4 V_{hu} L \quad (1)$$

where $u \in \{HD, LD\}$, $\alpha_1 V_{hu}$, $\alpha_2 V_{hu}$, $\alpha_3 V_{hu}$, and $\alpha_4 V_{hu}$; are

the coefficients associated with the vehicle type u . The cost value of VFCR (C_{ijk_u}) for a vehicle type u traveling from i^{th} site to j^{th} site in the city in k^{th} time-period is computed by adding first three terms in Eq. (1). Additionally, VFCR due to the vehicle's travel load (L) is calculated using the last term in Eq. (1) such as $\alpha_4 V_{hu} L_{ij}$, where L_{ij} is the travel load vehicle Vh when it is traveling from site i to site j .

2) *Carbon Emission*: The carbon emission depends on the rate of the vehicle fuel consumption (Eq. 1). Accordingly, the objective function of carbon emission is formulated as depicted in Eq. (2) by substituting Eq. (3) and Eq. (4) [18],[27]. The terms, VH : set of all vehicles; A : set of routes formed by all pairs of city sites (i,j) such that $(i \neq j)$; U : set of all vehicle types; M : set of all time-periods; N_u : maximum payload of vehicle type u ; C_{ijk_u} : cost of fuel consumption;

$\div V_{hu}$: $\{0, 1\}$ value indicating whether vehicle Vh belongs to type u ; D_{ij} : total distance traveled between source site i and destination site j ; $d_{ijk_{Vh}}$: continuous variable indicating the traveled distance of route $(i;j)$ in period k by vehicle Vh .

$$CO_2 = \sum_{Vh \in VH} \sum_{(i,j) \in A} \sum_{u \in \{U\}} CEC_1 + CEC_2 \quad (2)$$

$$CEC_1 = \sum_{k \in M} N_u C_{ijk_u} \div V_{hu} d_{ijk_{Vh}} \quad (3)$$

$$CEC_2 = N_u \div V_{hu} L_{ij} D_{ij} \quad (4)$$

In Eq. (2), the first term accounts for the carbon emissions produced by the vehicles themselves, while the second term represents emissions attributed to the loads carried by the vehicles. To facilitate flexible scheduling and enable fuel-saving or CO_2 emission reduction during traffic congestion, the continuous variable $d_{ijk_{Vh}}$ serves as the primary decision variable. It allows the route $(i;j)$ to be partitioned into multiple segments that can be traveled at different periods. This approach permits the possibility of discontinuous travel, where the vehicle can be temporarily off the road to mitigate emissions while navigating through congested traffic.

3) *Effective Incorporation of Time Window*: Travel time ($T_{ijk_{Vh}}$) respective to a vehicle (Vh) from i^{th} site to j^{th} site for k time-period, is computed in minutes using Eq. (5) which is linear because travel speed (S_{ijk}) is a parameter considered in km/h. Additionally, the total travel time of vehicle (Vh) in period k is restricted by applying time-bound via Eq. (6), wherein E_k and B_k are the ending and beginning time, respectively, of a trip $(i;j)$.

$$T_{ijk_{Vh}} = 60 \times \frac{d_{ijk_{Vh}}}{S_{ijk}} \quad A(i,j) \in A, k \in M, Vh \in VH \quad (5)$$

$$\sum_{(i,j) \in A} T_{ijk_{Vh}} \leq E_k - B_k \quad A \in M, Vh \in VH \quad (6)$$

The time window incorporation is constrained with the following Eqs. (7-11), wherein, Γ_i and Γ_j are the departures from i^{th} site and arrival time at j^{th} site, respectively; x_{ijkVh} is a binary variable indicating whether route (i,j) is traveled in period k by vehicle Vh (1) or not (0); X_{ij} is a binary variable indicating whether route (i,j) is traveled (1) or not (0); and $[\Delta^B, \Delta^E]$ represents start and end time constituting a time window to serve i^{th} ride service.

$$\Gamma_i^k \leq E_k - T_{ijkVh} + E_m \times (1 - x_{ijkVh}) \tag{7}$$

$$\Gamma_j^a \leq B_k + T_{ijkVh} - E_m \times (1 - x_{ijkVh}) \tag{8}$$

$$\Gamma_j^a \leq \Gamma_i^l + T_{ijkVh} - E_m \times (1 - X_{ij}) \quad A(i,j) \in A \tag{9}$$

$k=1 \quad h=1$

$$\Gamma_i^a + \Gamma_j^s \leq \Gamma_i^l \quad A_i \in \{1, \dots, n\} \tag{10}$$

$$\Delta^B \leq \Gamma_i^a \leq \Delta^E \quad A_i \in \{1, \dots, n\} \tag{11}$$

Constraint (Eq. 7) guarantees that if route (i,j) is traveled in period k by vehicle Vh (indicated by $x_{ijkVh}=1$), site i must be departed from before the period's end time minus the travel time of the route, where, $A(i,j) \in A, k \in M, Vh \in V, H$. Constraint (Eq. 8) ensures that the arrival time at node j must be greater than or equal to the period's start time plus the travel time of the route (i,j) . Constraint (Eq. 9) calculates earliest arrival times, active only for selected arcs (x_{ij}) . Constraint (Eq. 10) guarantees the vehicle departs after serving a node, considering service time. Finally, the constraint (Eq. 11) sets the bounds on arrival time at customer node i . Together, these constraints manage time-related aspects of the vehicle routing problem, ensuring timely service and adherence to time windows.

4) **Travel Distance:** The objective function for travel distance (TD) is designed to enforce the maximum travel distance limits of vehicles based on their fuel tank capacity (F_u), is formulated in Eq. (12) and Eq. (13), where, $Vh \in VH$. It ensures that the total amount of fuel consumed by vehicle Vh (second term) does not exceed its fuel tank capacity (first term). This is crucial to prevent vehicles from traveling beyond their fuel constraints and helps in maintaining efficient and feasible routes for the vehicle routing problem. By optimizing the objective function TD , the solution aims to minimize travel distance while satisfying these fuel constraints for all vehicles in the fleet.

5) **Traffic Congestion:** The traffic congestion model associated with a specific vehicle (Vhu) is formulated using Eqs. (14-19). At an instance t , fuel consumption rate ($FCVhu$), carbon emission (CO_2), travel time (T_{ijkVhu}), travel distance ($TDVhu$), and traffic condition on route ($d_{ijkVh} \times \tau_{ijkVh}$) respectively of vehicle (Vh^u) decides its traffic congestion status $status_{Vhu}$ based on the corresponding threshold parameters'

($status_{Vhu}$) based on the corresponding threshold parameters'

$$TD = \sum_{(i,j) \in A} D_{ij} \times X_{ij} \quad \text{subject to} \tag{12}$$

$$F_u \geq \sum_{u \in U} \sum_{(i,j) \in A, k \in K, u \in U} C_{iku} d_{ijkVh} + \sum_{u \in U} L_{ijVh} d_{ij} \tag{13}$$

$$\text{values } \{ \omega_{Vh}^{FC}, \omega_{Vhu}^{FC}, \omega_{Vhu}^{CO_2}, \omega_{Vhu}^T, \omega_{Vhu}^{TD} \}.$$

$$\textcircled{1} Vh^u = \textcircled{1} Vh^u + \textcircled{1} Vh^u + \textcircled{1} Vh^u + \textcircled{1} Vh^u + \textcircled{1} Vh^u \quad (14)$$

$$\textcircled{1} Vh^u = FC_{Vh^u} \times \omega_{Vh^u}^{FC} \quad (15)$$

$$\textcircled{1} Vh^u = CO_2_{Vh^u} \times \omega_{Vh^u}^{CO_2} \quad (16)$$

$$\textcircled{1} Vh^u = T_{ijkVh^u} \times \omega_{Vh^u}^T \quad (17)$$

$$\textcircled{1} Vh^u = TD_{Vhu} \times \omega_{Vh^u}^{TD} \quad (18)$$

$$\textcircled{1} Vh^u = Vh^u \times d_{ijkVh^u} \times \tau_{ijkVh^u} \times \omega_{Vh^u}^T \quad (19)$$

Vh

C. Problem Statement

Specifically, the considered problem discussed above, is a Multiple objectives constrained crowd delivery vehicle traffic trade-offs among them. First and foremost, we must adhere to strict travel time windows $[\Delta^B, \Delta^E]$ to ensure timely deliveries or services at customer locations. Concurrently, minimizing the travel distance (TD) is of utmost importance, as it not only reduces operational costs but also curtails fuel consumption (FC) and carbon emissions (CO2), aligning with sustainability goals. Nevertheless, optimizing travel distance must account for potential traffic congestion, as congested routes can lead to delays and inefficiencies. Accordingly, the problem can be mathematically stated as Eq. (20) subject to constraints specified in Eq. (21).

$$\begin{aligned} & \times \\ & \textcircled{1} OR_i \\ & i=i \\ & = \text{MIN}(FC_{Vh^u}(S, L); CO_2_{Vh^u}; T_{ijkVh^u}; TD_{Vh^u}, \textcircled{1} Vh^u) \\ & \text{s.t. } \{C_1 - C_7\} \end{aligned} \quad (20)$$

$$\begin{aligned} C_1: & X_{ij} + X_{ji} \leq 1 \quad A(i, j) \in A \\ C_2: & X_{ij} \geq y_{ijVh} \quad A(i, j) \in A, Vh \in VH \\ C_3: & X_{ij} \geq x_{ijkVh} \quad A(i, j) \in A, Vh \in VH, k \in M \\ C_4: & y_{i0Vh} \leq 1 \quad Vh \in VH \\ C_5: & \sum_{i=0} Vh^u C_u \geq \sum_{i=0} Vh^u \quad \sum_{j=1} Vh^u y_{ijVh} \\ C_6: & 1 \leq \sum_{i=1} X_{i0} \leq q \\ C_7: & 1 \leq \sum_{j=1} X_{0j} \leq q \end{aligned} \quad (21)$$

C: One route should be traveled at most once; C: If a

1

2

oute (i,j) is not selected, all y_{ijVh} should be zero; C3: If

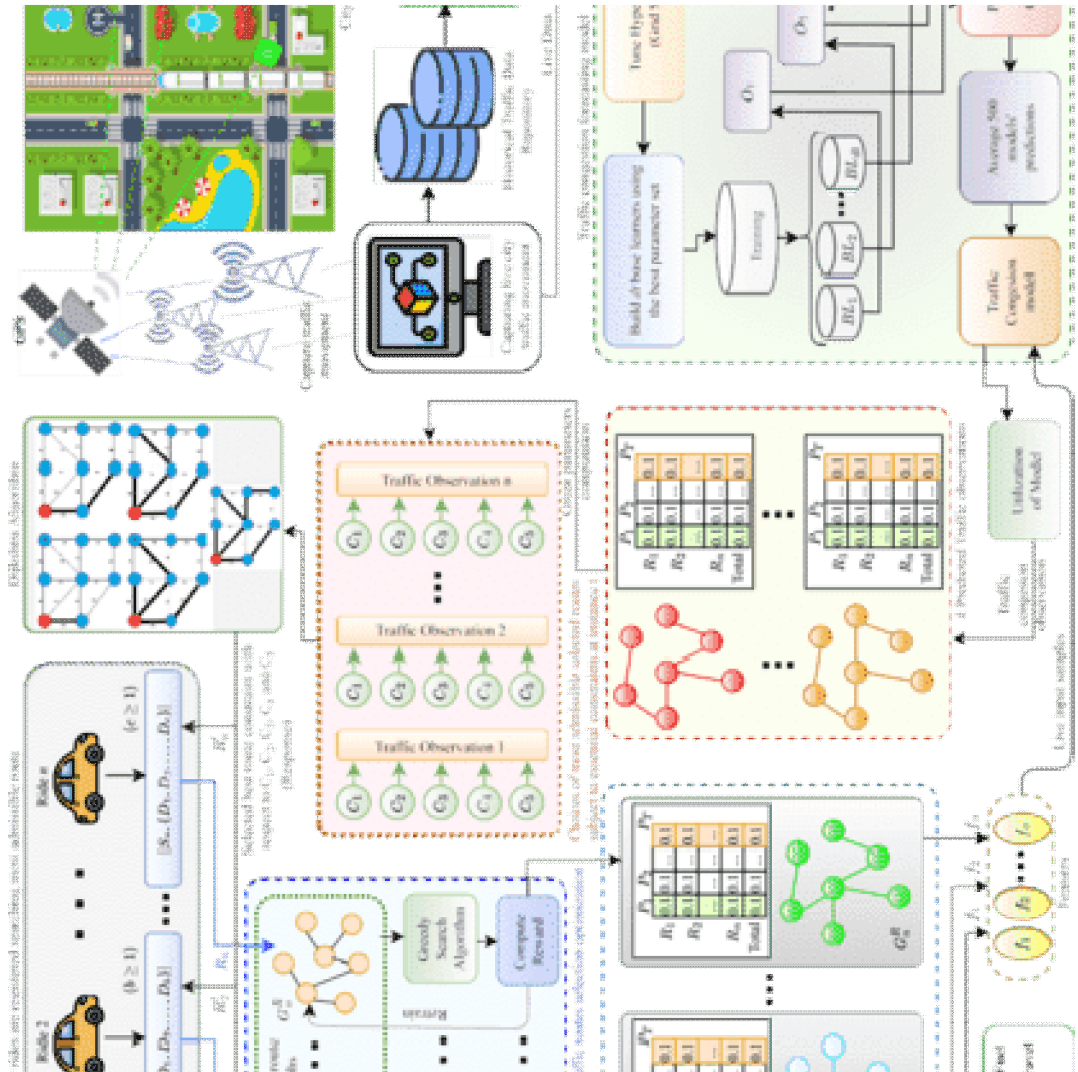


Fig.1. Real Time Traffic Condition Based Smart Signal System.

a route (i, j) is not selected, all $x_{ijk}Vh$ should be zero; $C4$: Each vehicle returns to depot only once; $C5$: load capacity of a vehicle should not be violated; $C6$: total number of returns to the depot is in $[1, q]$; $C7$: total number of departures from the depot is in $[1, q]$, q : total number of vehicles.

III. IM-VRMMODEL

Let each of the single-depot or multi-depot delivery ride node $v \in V$ is associated with features x_v , that describe its specific attributes.

requests $\{R_1, R_2, \dots, R_n\}$ is represented as a GNN to

capture spatial relationships between locations as illustrated in Fig. 1. For each ride, the number of possible travel routes covering multiple depot sites starting from the source site, are generated using the information about locations viz.,

latitude (ϕ) and longitude (λ) of the source site and depot sites which allows computation of the distance between two locations (v and v_j) using *haversine function* ($hav(\vartheta)$) via following mathematical derivation:

Let the *central angle* (ϑ) between any two points on a sphere be computed using Eq. (22), where d is the distance between the two points on the sphere, and r is the radius of the sphere.

Here, l represents the current layer of the GNN, $N(v)$ denotes the set of neighboring nodes of v , and $h^{(l)}$ refers to the feature representation of node v at layer l . The function AGGREGATE collects and consolidates information from the neighboring nodes, while COMBINE merges the aggregated information with the node's existing features. These equations are applied iteratively across multiple layers, allowing the GNN to capture complex dependencies and relationships within the graph structure.

A. Travel Route Network Optimization

$$d = r \vartheta \quad (22)$$

Then GNNs $\{G_1, G_2, \dots, G_n\}$ are optimized with a specific objective, i.e., minimizing distance. This objective is

The *haversine* of ϑ ($hav(\vartheta)$) is computed directly from the latitude (ϕ) and longitude (λ) of the source site and depot sites using *haversine formula* as stated in Eq. (23), where, ϕ_1, ϕ_2 are latitude of point 1 and point 2, respectively, and λ_1, λ_2 are longitude of point 1 and point 2, respectively.

$$hav(\vartheta) = hav(\phi_2 - \phi_1) + \cos(\phi_1) \cos(\phi_2) hav(\lambda_2 - \lambda_1) \quad (23)$$

Applying *haversine function* ($hav(\vartheta)$) to both the *central angle* (ϑ) and differences in latitude and longitude is given by Eq. (24):

$$hav(\vartheta) = \sin^2 \frac{\vartheta}{2} = \frac{1 - \cos(\vartheta)}{2} \quad (24)$$

haversine to $h = hav(\vartheta)$ as stated in detail using Eq. (25):

This model incorporates GNNs, which are specifically designed to effectively model graph-structured data. These networks are particularly well-suited for tasks that involve relational and spatial information. GNNs leverage a mechanism known as message passing, where nodes iteratively exchange and aggregate information from their neighbors. This enables the network to learn rich and expressive representations of graph elements while preserving the underlying structural dependencies. In this model, the travel routes are represented as an input graph $G(V, E)$, where V denotes the set of nodes representing locations. These locations include the source site

node $v \in V$ is associated with features x_v that describe its specific attributes.

To update these node features, the GNN employs a layered message-passing mechanism. At each layer, the node features are updated iteratively using the following Eqs. (26 and 27):

$$h^{(l+1)} = \text{AGGREGATE}\{h^{(l)} \mid u \in N(v)\} \quad (26)$$

$$h^{(l+1)} = \text{COMBINE}(h^{(l)}, h^{(l+1)}) \quad (27)$$

incorporated by adding a loss function (L_f) that measures the error between the predicted values (e.g., distances) and the true values (ground truth distances). Let \hat{d} denote the predicted distance between nodes v_1 and v_2 as $\hat{d}(v_1, v_2)$ and the true distance as $d(v_1, v_2)$.

The loss function L_f for the GNN can be defined as a sum of squared errors over all edges in the graph as stated in Eq. (28):

$$L_f = \sum_{(u,v) \in E} (\hat{d}(v_1, v_2) - d(v_1, v_2))^2 \quad (28)$$

The GNN's parameters are adjusted to minimize this loss using the Gradient Descent algorithm. Thereafter, GNN is utilized to compute node embeddings that encode information about the locations and their relationships. These embeddings can

The *haversine function* computes the *halfaversine* of the angle

ϑ . The distance d is computed using the *inverse haversine* (*inverse* be extracted from the final GNN layer's node representations by applying

Eq. (29), where θ is the final layer of the GNN.

$$Embedding(v) = h^{(\theta)} \tag{29}$$

$$d = r \cdot \arccos(\frac{h \cdot h'}{\|h\| \|h'\|}) = 2r \cdot \arcsin(\sqrt{1 - \frac{h \cdot h'}{\|h\| \|h'\|}})$$

The Greedy Search (GS) algorithm operates with the initial and multiple depot sites, given by $S^i, \{D^1, D^2, \dots, D^a\}$,

where $1 \leq i \leq n$. These E represents the edges, which

initialization of a starting node (e.g., the starting point of your route), and then node embeddings obtained from the GNN.

GS algorithm iteratively selects the next node based on the smallest distance in the embeddings space, denoted as Eq. (30), where, N/R represents the set of unvisited nodes, and $Embedding(v)$ represents the embedding of node v obtained from the GNN.

$$v = \arg \min_{v \in N/R} Embedding(v) \tag{30}$$

The GS optimization is continued until all nodes have been visited or a termination condition is achieved. The integration of GNNs and the GS algorithm enables leveraging of the spatial relationships captured by the GNN to guide the corresponding potential routes between these locations. Each selection of nodes during route optimization.

A. Handling Traffic Congestion With Green Parameters Optimization

Efficient vehicle traffic management is achieved through the regularization term, denoted by W in Eq. (35), prevents the model from overfitting by applying penalties to the model's complexity:

Traffic Congestion Forecasting Model unit (TCFMu), which is designed to handle dynamic traffic conditions through periodic

$$L_t = \sum_{i=1}^n (X^s - O^s)^2 + L(O^s, O^{s'}) + BL(F) + W(BL^s)$$

(34)

training and retraining. The training process is initiated using

$$i=1, t=1, \dots, t$$

historical traffic data, represented as the set $\{ds_1, ds_2, \dots, ds_m\}$,

sourced from the *Historical Traffic Data Repository* (kd-HTDR). To ensure high data quality, preprocessing techniques

The regularization term is defined as:

$$W(BL^s) = \gamma K + \frac{1}{\|w\|^2} \tag{35}$$

are applied, including handling missing values (e.g., through

mean imputation), and preparing the dataset for both training and testing purposes.

Following preprocessing, *Feature Selection* identifies the most relevant features for predicting traffic congestion. The *Pearson correlation coefficient* (P^{Corr}), which measures the linear relationship between continuous variables, is used.

It ranges from -1 (perfect negative) to $+1$ (perfect positive), with values near 0 indicating no relationship. The formula for

P^{Corr} is given by Eq. (31), where, x_i and y_i are the values of the characteristic and target variables at the data point i , with \bar{x} and \bar{y} as their respective means.

P_z

where γ and λ are regularization parameters, K is the number of leaves in the decision tree, and w represents the weights of the tree splits.

This iterative model enhancement helps to optimize both predictive performance and model complexity by gradually correcting errors, incorporating more data, and carefully balancing model bias and variance.

After the model is trained, it is applied to live data, such as traffic data from the *Live City Traffic* (kd-LCT) database, to predict traffic congestion for upcoming time intervals.

Once the model generates congestion predictions, high-congestion routes are discarded, and *Dijkstra's algorithm* is used for route optimization. This algorithm identifies the

$pCorr$

$$= \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}}$$

(31) shortest path in a graph, accounting for predicted congestion levels and various *Green Parameters* such as vehicle

$$\sum_{i=1}^n (x_i - \bar{x})^2 \quad \sum_{i=1}^n (y_i - \bar{y})^2$$

fuel consumption, carbon emissions, time windows, and

After feature selection, *Feature Encoding* transforms categorical variables into numerical values using *Label Encoding*, assigning each unique category a distinct integer. The dataset is then split into training and testing subsets. For modeling, *Extreme Gradient Boosting* (XGBoost), an efficient gradient boosting method, trains the Traffic Congestion Forecasting Model (TCFMu). XGBoost constructs an ensemble of decision travel distance. The *Green parameters* are computed using Eqs. (36- 39):

$$\text{FuelConsumption(FC)} = \sum_{i=1}^n FC_i V_h(S, L) \quad (36)$$

$$\text{CarbonEmission(CE)} = \sum_{i=1}^n CO2_i \quad (37)$$

trees, iteratively adding new trees to correct previous errors, minimizing an objective function that combines prediction accuracy and regularization at each step. To evaluate the performance of TCFMu, we use the *Mean Squared Error* (MSE) score, calculated using Eq. (32), where m is the total

TravelTime=

$$\sum_{i=1}^n \sum_{j=1}^n T_{ijk} V_h \quad (38)$$

number of data points, and D^{Ac} and D^{Pr} represent the actual

TravelDistance=

$$\sum_{i=1}^n (D_i^{Ac} - D_i^{Pr})^2 \quad (39)$$

and predicted data for data point i , respectively:

$$MSE = \frac{1}{2} \sum_{i=1}^n (D_i^{Ac} - D_i^{Pr})^2 \quad (32)$$

By carefully optimizing these green parameters, we aim to balance traffic efficiency with environmental sustainability.

$$m \quad i \quad i \quad i=1$$

XGBoost enhances predictive power by building the model incrementally with decision trees, minimizing the loss function. The model consists of a set of base learners (decision

trees) denoted as $BL^z = \{BL^1, BL^2, \dots, BL^m\}$, which output predictions as stated in Eq. (33), where, m is the number of decision trees:

$$O^z = \sum_{i=1}^m BL_z^i(F_i) A_i \in \{1, 2, \dots, s^z\} \quad (33)$$

During each iteration, the algorithm attempts to minimize the objective function, which is composed of two terms: a loss term, which measures the prediction error, and a regularization term, which penalizes the complexity of the model. The loss term, denoted by L in Eq. (34), quantifies the difference between the actual and predicted values, while

IV. OPERATIONAL DESIGN AND COMPLEXITY

The IM-VRM model, as outlined in Algorithm 1, utilizes three primary knowledge databases: Live City Traffic ($kd-LCT$), Vehicle Routes Connecting City ($kd-VRCC$), and Historical Traffic Data Repository ($kd-HTDR$). Customers generate multiple depot delivery ride requests in each time interval

$t, t + \Delta t$ (Step 1), which triggers the system to create travel route graphs (TRGs) for each request (Step 2). Each TRG

undergoes optimization via a Greedy Search algorithm that generates a reward score based on travel time and cost (Step 3). The system then predicts congestion on the travel route using the XGBoost-based Traffic Congestion Forecasting Model (Step 4). If no congestion

is detected (Step 5), green parameters such as vehicle fuel consumption ($FC_{V_{jk}^{V_{nu}}}$), carbon emissions ($CO_{2V_{jk}^{V_{nu}}}$), travel time ($T_{ijk}^{V_{nu}}$), and distance ($TD_{ijk}^{V_{nu}}$) are calculated, and Dijkstra's algorithm is applied to select the most admissible route based on these green

constraints (Step 6). If congestion is predicted for a route (Step 7), the system re-schedules the route and generates an alternative path to continue the process. Thus, the IM-VRM model ensures optimized and environmentally efficient vehicle routing in dynamic traffic conditions.

Algorithm 1 IM-VRM: Operational Summary

```

1 Input: Three knowledge databases: Live City Traffic (kd-LCT), Vehicle Routes Connecting City (kd-VRCC), and Historical Traffic Data Repository (kd-HTDR);
2 for each time interval  $\{t, t + \Delta t\}$  do
3   Step 1: Customers generate  $m$  multi-depot delivery ride requests  $\{R_1, R_2, \dots, R_m\}$ ;
4   for each ride  $\{R_1, R_2, \dots, R_m\}$  do
5     Step 2: Generate travel route graphs (TRGs)  $\{G^1, G^2, \dots, G^m\}$ , where nodes  $i \in \{1, \dots, n\}$  represent depot and sites;
6     for each TRG  $\{G, G, \dots, G\}$  do
7       Step 3: Use the Greedy Search algorithm to optimize and generate the 'reward score' iteratively, considering travel time and cost for each route;
8       Step 4: CALL XGBoost algorithm-based Traffic Congestion (⊕) Forecasting Model (TCFMu) to forecast travel route congestion status  $\sigma_{status}$  for the respective node;
9       if  $\sigma_{status} = FALSE$  then
10         Step 5: Compute the cost of green parameters:  $FC_{V_{jk}^{V_{nu}}}$ ,  $CO_{2V_{jk}^{V_{nu}}}$ ,  $T_{ijk}^{V_{nu}}$ , and  $TD_{ijk}^{V_{nu}}$  using Eqs. (36-38) and Eq. (39), respectively;
11         Step 6: Apply Dijkstra's algorithm to select the most admissible travel route constrained by green parameters;
12       else
13         Step 7: Re-schedule the TRG with an alternative route and continue;

```

Time complexity: The time complexity of the IM-VRM model is composed of several key components. First, in Line 2, the model iterates over Δt time intervals, contributing a complexity of $O(\Delta t)$. In Step 1, generating multi-depot ride requests has a constant time complexity of $O(1)$. Then, in Step 2, processing n rides results in a complexity of $O(n)$. For processing TRGs, Step 3 involves Greedy optimization, which has a constant complexity of $O(1)$. Step 4 applies the XGBoost-based Traffic Congestion Forecasting Model

(TCFMu), contributing a complexity of $O(t^\dagger \cdot d^\dagger)$, where t^\dagger and d^\dagger are the number of trees and their depth, respectively.

The complexity for processing the travel route graphs (TRGs) using GNN is $O(Z \cdot (N + E))$, where Z is the number of layers, N is the number of nodes, and E is the number of edges. Step 5, which calculates green parameters, and Step 7

re-scheduling the TRGs, both have constant complexity of $O(1)$. Step 6, where Dijkstra's algorithm is applied for route

selection, contributes a complexity of $O((N + E) \cdot \log N)$. Combining all these factors, the overall time complexity of the

IM-VRM model is expressed as: $O(\Delta t \cdot n \cdot t^\dagger \cdot d^\dagger + Z \cdot (N + E) + (N + E) \cdot \log N)$. This demonstrates the scalability and efficiency of the model for real-time, large-scale operations

V. PERFORMANCE EVALUATION

A. Experimental Setup

The experimental work is carried out on a server machine comprising two Intel® Xeon® Silver 4114 CPU with a 40 core processor and having 2.20 GHz clock speed. The simulation machine runs on Ubuntu 16.04, an 64-bit LTS operating system comprising 128 GB of main memory. Enactment of the proposed work is carried out using Python 3.9 over an extended version of the T-Drive trajectory dataset [28].

B. Implementation

The design and operational flow of the proposed model is implemented and evaluated in three consecutive steps as depicted in Fig. 2. In essence, the IM-VRM model is the result of collaborative work among different modules as follows:

- *ride initialization():* The users generate requests for multi-depot rides i.e. crowd deliveries. These rides can be initialized for different vehicles categorized into three types heavy, middle, and lightweight.
- *constraints:* Constraints play a crucial role in shaping the effectiveness of the IM-VRM model. Each ride is subject to specific constraints, such as ensuring depots aren't repeated, the origin and destination are distinct, and rerouting in case of congestion, among others.
- *initial route selection():* GNN employing a greedy optimization approach is deployed to generate an initial route for a ride. These routes are further processed for countering traffic congestion and green parameters. Table II showcases the parameter value utilized for the model in this step.
- *traffic congestion prediction():* The system predicts potential traffic congestion on the initially chosen route by using a combination of live and historical data stored in its knowledge base. Table II depicts the parameter details utilized for the model training at this step.
- *Time complexity:* The time complexity of the IM-VRM model is composed of several key components. First, in Line 2, the model iterates over Δt time intervals, contributing a complexity of $O(\Delta t)$. In Step 1, generating multi-depot ride requests has a constant time complexity of $O(1)$. Then, in Step 2, processing n rides results in a complexity of $O(n)$. For processing TRGs, Step 3 involves Greedy optimization, which has a constant complexity of $O(1)$. Step 4 applies the XGBoost-based Traffic Congestion Forecasting Model (TCFMu), contributing a complexity of $O(t^\dagger \cdot d^\dagger)$, where t^\dagger

and d^\dagger are the number of trees and their depth, respectively.

The complexity for processing the travel route graphs (TRGs) using GNN is $O(Z \cdot (N + E))$, where Z is the number of layers, N is the number of nodes, and E is the number of

edges. Step 5, which calculates green parameters, and Step 7,

- *knowledge database*: A data repository comprising live traffic details captured from Global Positioning System (GPS) and historical details from different data repositories. Data accumulation is an ongoing and continuous process to capture data for further analysis.
- *kd-LCT*: acknowledged database for Live City Traffic
- *kd-VRCC*: acknowledged database for Vehicle Routes Connecting City
- *kd-HTDR*: acknowledged database for Historical Traffic
- *green parameters()*: Major green parameters for the IM-VRM model include minimizing carbon emissions, reducing fuel consumption, lowering fuel costs, optimizing route distances, and minimizing travel time.

TABLE III DATASET DESCRIPTION

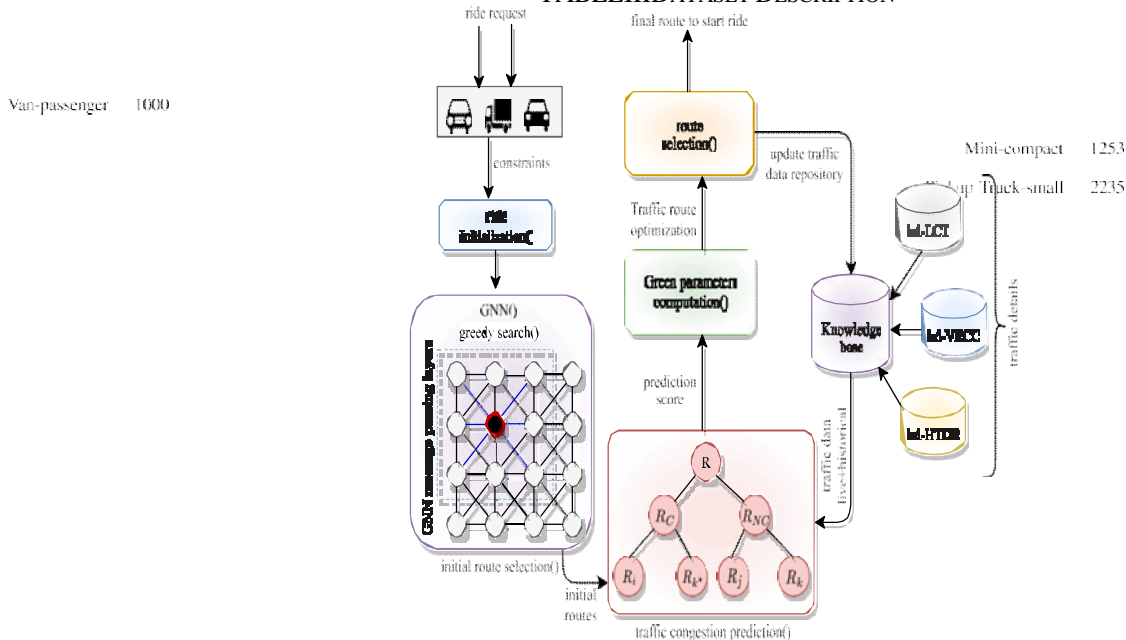


Fig.2.Implementation workflow for IM-VRM model.

TABLE III EXPERIMENTAL PARAMETERS

GNN parameters setup for initial route selection			
Parameter	Value	Parameter	Value
Learning rate	3e-4	Entropy value	0.01
Hidden node dimension	128	Eps clip	0.2
Hidden edge dimension	16	Time step	1
Epoch	100	ppo epoch	3
Batch size	512	Data size	512000
Convolution layers	4	Validation size	10000
XGBoost parameters setup for traffic congestion prediction			
Parameter	Value	Parameter	Value
Base score	0.05	Learning rate	0.5
Booster	gbtree	Max delta step	0
Colsample_bylevel	1	Max depth	5
Colsample_bynode	1	Min child weight	1
Colsample_bytree	1	n_estimators	1200
Gamma	0	objective	multi:softprob
Seed	5	Verbosity	1

- *optimal routes selection()*: Utilizing the Dijkstra algorithm, an optimal ride is planned by considering relevant green parameters and congestion prediction value.

C. Dataset

The performance of the proposed model is evaluated using an extended T-Drive trajectory dataset of Beijing city which contains a one-week trajectories of 10,357 taxis. The dataset consists of approximately 15 million data points and covers trajectories spanning a total distance of about 9 million kilometers [28]. For the extended dataset, there are over 40,000 training samples. Each sample includes ten features for each vehicle, such as longitude, latitude, timestamp, vehicle type, manufacturer, model, vehicle class, engine size, fuel type, cylinders, and transmission. Some features, such as distance, speed and time to travel, are derived from the above attributes to assess traffic congestion for route optimization based on green parameters. The traffic conditions are categorized as: 'no-traffic,' 'mild-traffic,' and 'heavy-traffic.' The details of the range of parameters for the extended dataset and information on vehicle categories and their respective fuel types are given in Table III.

TABLE III DATASET DESCRIPTION

Features description			
Feature	Range	Feature	Range
Speed	0 to 89.104 km/h	Date	02-02-2008 to 08-08-2008
Distance	0 to 29.14 km	Time	13:33:02 to 17:39:15
Time	0 to 1.54 hour	Longitude	115.0736 to 117.43605
CO ₂ Emission	0 to 8.090 kg	Latitude	38.81899 to 42.35342
Fuel Consumption	0 to 4.268 litre		
Vehicle types description			
Vehicle	Name	# Instances	# VFI
Heavy (16224)	Pickup Truck-standard	15224	E = 0 D = 3003 X = 13324 C = 0
	Van-cargo	1000	
	Van-passenger	1000	E = 1005 D = 0
Middle (4488)	Mini-compact	1253	X = 2230 C = 1253
	Pickup Truck-small	2235	
Light (22550)	SUV-small	15802	E = 1005 D = 3451
	SUV-standard	6748	X = 17107 C = 1992

E: Electric, D: Diesel, X: Petrol, C: CNG, # VFI: Number of vehicles of specific fuel types

D. Performance Parameter Analysis

1) *Initial Travel Route Selection*: Table IV presents the outcomes generated by employing a Greedy approach optimized GNN for the initial route selection in crowd delivery rides. The primary goal of this approach is to choose a route with the minimum travel distance while ensuring the successful delivery of parcels to their intended destination depots.

Notably, as the learning process unfolds across multiple epochs, the average travel distance progressively diminishes. This trend signifies the efficacy of the applied GNN optimization technique. Our experiments spanned a range of epochs, from 10 to 100, with the most favorable results emerging after thirty epochs, as depicted in Table IV. These results encompass comprehensive insights into the rewards and loss points observed during successive batches over the course of the thirty-epoch learning period. The rewards fall within

the [2.9-3.9] range, while the loss points range from [—0.01-3.6]. An intriguing observation is the declining P-value over

batches within consecutive epochs. This trend suggests that the outcomes obtained are not mere chance occurrences; rather, they indicate a statistically significant relationship or effect within the experimental data.

The experiments encompassed a diverse set of multi-depot delivery rides. However, due to space constraints, in Fig. 3, we present the initially selected routes resulting from GNN optimization for a subset of nine multi-depot delivery rides. To clarify the graph's content, the X-axis and Y-axis represent the scaled latitudinal and longitudinal coordinates of the respective depots in each ride. In these graphical representations, the red bold dot serves as the starting point for the multi-depot ride, while the green dots symbolize multiple delivery depots or drop-off points. In this particular visualization, we focus on rides with precisely eight delivery depots. These routes represent the outcomes of a greedy optimization approach aimed at minimizing travel distance using GNN. Notably, these optimizations do not take into account additional constraints such as traffic congestion, carbon emissions, fuel consumption, and other environmentally

TABLE IV
INITIAL TRAVEL ROUTE OPTIMIZATION FOR A RANDOM MULTI-DEPOT RIDE

Epoch	Batch	Reward	Loss	Execution Time	P-value	Avg. Distance
Epoch 0	17100	3.7740	-3.5047	1.3016	6.0	-
	517100	3.2334	-3.5531	1.2513		
	997100	3.2360	-3.3253	1.2285		
Epoch 1	17100	3.1970	-3.5888	1.2480	4.6017	-
	517100	3.0920	-3.1190	1.2353		
	997100	3.1210	-3.1692	1.2609		
Epoch 2	17100	3.1060	-3.3881	1.2463	7.6766	3.6409
	517100	3.0650	-3.3138	1.2355		
	997100	3.0700	-3.1318	1.2439		
Epoch 3	17100	3.0590	-3.1648	1.2609	5.8369	3.0214
	517100	3.0530	-3.3091	1.2367		
	997100	3.0410	-3.0789	1.2539		
Epoch 4	17100	3.0360	-3.1130	1.2652	3.8271	3.0046
	517100	3.0070	-3.1192	1.2399		
	997100	3.0436	-3.2038	1.2378		
Epoch 5	17100	3.0370	-3.2522	1.2358	0.0001	2.9951
	517100	2.9970	-3.1253	1.2319		
	997100	3.0270	-3.06590	1.2438		
Epoch 6	17100	3.0170	-3.1923	1.2325	7.0902	2.9951
	517100	2.9760	-3.1055	1.2336		
	997100	3.0170	-3.1231	1.2362		
Epoch 7	17100	3.0070	-3.1557	1.2053	0.1070	2.9798
	517100	2.9770	-3.1306	1.2393		
	997100	3.0100	-3.1448	1.2676		
Epoch 8	17100	2.9980	-3.1577	1.2943	1.3652	2.9660
	517100	2.9610	-3.08730	1.2591		
	997100	3.0030	-3.11240	1.2899		
Epoch 9	17100	2.9810	-3.1854	1.2920	0.6221	2.9599
	517100	2.9880	-3.1599	1.2866		
	997100	2.9830	-3.0717	1.2695		
Epoch 10	17100	2.9590	-3.1264	1.3008	4.7907	2.9469
	517100	2.9520	-3.1657	1.2701		
	997100	2.9830	-3.1914	1.2733		
Epoch 11	17100	2.9570	-3.0898	1.3094	0.1379	2.9442
	517100	2.9480	-3.1599	1.3344		
	997100	2.969	-3.0862	1.3260		
Epoch 12	17100	2.9510	-3.0910	1.3215	3.9143	2.9369
	517100	2.9300	-3.0899	1.2739		
	997100	2.9650	-3.1096	1.2817		
Epoch 13	17100	2.9520	-3.1596	1.4045	0.0006	2.9310
	517100	2.9230	-3.0830	1.3407		
	997100	2.9560	-3.0803	1.4164		
Epoch 14	17100	2.9335	-3.0974	1.3834	2.9332	
	517100	2.9280	-3.0888	1.3696		
	997100	2.9640	-3.1192	1.2999		
Epoch 15	17100	2.9341	-3.1226	1.3438	-	2.9335
	517100	2.932	-3.2317	1.3089		
	997100	2.9720	-3.1685	1.2777		
Epoch 16	17100	2.943	-3.1094	1.3223	0.0724	2.9283
	517100	2.937	-3.1764	1.3693		
	997100	2.9410	-3.0934	1.3555		
Epoch 17	17100	2.9370	-3.0788	1.3852	0.0069	2.9269
	517100	2.9320	-3.1764	1.3694		
	997100	2.9520	-3.0852	1.3097		
Epoch 18	17100	2.9311	-3.2024	1.2928	0.0722	2.9249
	517100	2.9170	-3.0648	1.2916		
	997100	2.9470	-3.0877	1.3603		
Epoch 19	17100	2.9310	-3.1054	1.4664	0.0115	2.9233
	517100	2.9190	-3.0848	1.3937		
	997100	2.9440	-3.0973	1.3175		
Epoch 20	17100	2.9380	-3.1572	1.3329	-	2.9236
	517100	2.9090	-3.0498	1.3552		
	997100	2.9480	-3.1532	1.3055		
Epoch 21	17100	2.9290	-3.2828	1.3235	-	2.9236
	517100	2.9210	-3.0995	1.3117		
	997100	2.9450	-3.0504	1.3452		
Epoch 22	17100	2.9240	-3.0733	1.3348	0.0583	2.9209
	517100	2.9180	-3.0968	1.3109		
	997100	2.9340	-3.0844	1.2873		
Epoch 23	17100	2.9230	-3.0800	1.3463	0.1092	2.9217
	517100	2.9090	-3.0582	1.2970		
	997100	2.9330	-3.0472	1.3279		
Epoch 24	17100	2.9270	-3.0719	1.3753	0.0971	2.9213
	517100	2.9080	-3.0461	1.3429		
	997100	2.9460	-3.0579	1.3260		
Epoch 25	17100	2.9240	-3.0716	1.3270	-	2.9204
	517100	2.9080	-3.0521	1.3367		
	997100	2.9370	-3.0378	1.3818		
Epoch 26	17100	2.9150	-3.0547	1.3551	-	2.9205
	517100	2.910	-3.0731	1.2873		
	997100	2.940	-3.0606	1.3845		
Epoch 27	17100	2.9170	-3.0752	1.2582	0.1123	2.9188
	517100	2.911	-3.1033	1.3178		
	997100	2.932	-3.0482	1.2416		
Epoch 28	17100	2.9160	-3.0499	1.2767	0.2900	2.9197
	517100	2.9100	-3.0773	1.2439		
	997100	2.9350	-3.1026	1.2803		
Epoch 29	17100	2.9210	-3.0772	1.3133	0.2099	2.9193
	517100	2.9100	-3.0964	1.2797		
	997100	2.9350	-3.0568	1.3013		

conscious parameters. The total travel distance achieved during the initial route optimization is prominently displayed at the top of each graph, providing a clear reference for the efficiency of the selected routes in these multi-depot rides. Further, the outcome of travel routes optimization considering all the green parameters constraints are presented in subsequent sections.

2) *Traffic Congestion Prediction:* The experimental results pertaining to the accuracy of traffic congestion prediction for a range of multi-depot rides exhibit a gradual variation across different learning rates, falling within the intervals of [0.01-0.28] and [0.10-0.98]. These variations are visually represented in Fig. 4(a) and Fig. 4(b), respectively. In Fig. 4(a), it is observed that accuracy scores for traffic congestion prediction as they change with different learning rates within the range of [0.01-0.28]. Fig. 4(b) presents a similar depiction but focuses on the range [0.10-0.98]. Notably, the average accuracy score for predicting traffic congestion in various multi-depot rides reaches an impressive level of up to 92.5% across this spectrum of learning rates. The experimental results highlight a key observation: the deviation in prediction accuracy performance during both training and testing sessions is minimized when learning rates are within the range of [0.10-0.98]. In contrast, when learning rates fall within the range [0.01-0.28], the accuracy scores exhibit greater fluctuations.

This finding reflects the enhanced efficiency and robustness of accurate traffic prediction in scenarios where learning rates are set within the range of [0.10-0.98].

Moreover, Fig. 5 illustrates the accuracy scores obtained for a systematic range of minimum child weight values, ranging from 0.0 to 4.0, with increments of 0.5. These evaluations were conducted within the learning rate range of [0.10-0.98]. The purpose of these graphs is to showcase the consistent performance exhibited during both the training and testing phases of our proposed traffic congestion prediction approach, which leverages the XGBoost algorithm. In Fig. 6, a comprehensive summary of the average results is provided. It includes prediction accuracy, precision, recall, and F1 score, all achieved within the context of the learning rate falling within the [0.10-0.98] range. Impressively, the accuracy score reaches a peak of 92.5%, while precision, recall, and F1 score values reach as high as 0.93. This detailed analysis emphasizes the robust and dependable performance of traffic congestion prediction across varying minimum child weight values and highlights the significant predictive capabilities of XGBoost algorithm within the specified range of learning rate.

3) *Optimal Travel Route Constrained With Green Parameters:* In Fig. 7, the attained optimal delivery routes for a variety of multi-depot rides are presented in both zoomed-in and zoomed-out view. These routes are the result of a comprehensive optimization process that involves the utilization of GNN optimization, traffic congestion prediction, and the application of the Dijkstra algorithm to optimize green parameters. These green parameters include factors such as optimal fuel consumption, minimal carbon emissions, and the least time required for delivery. The routes depicted in the figure represent the most recommended and viable routes for each multi-depot ride. They take into account real-time traffic congestion conditions and are meticulously designed to achieve an optimal balance between fuel efficiency, environmental considerations (carbon emissions), and the efficient use of time. Fig. 8 and Fig. 9 provide a detailed illustration of the performance of the proposed model. These figures depict the computation of critical green parameters, including fuel consumption, carbon emissions, and distance traveled, under two distinct scenarios: one in the presence of traffic congestion

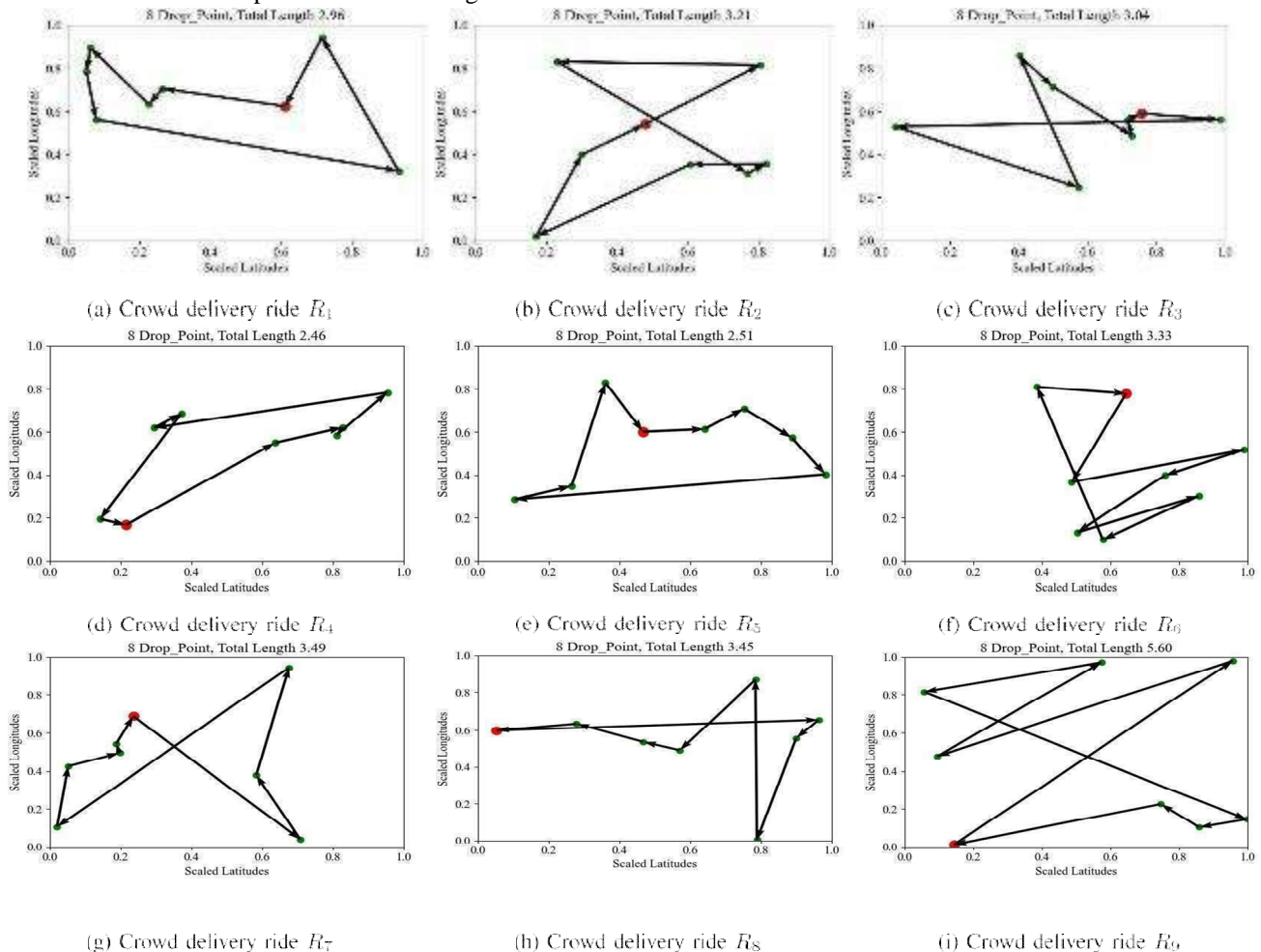
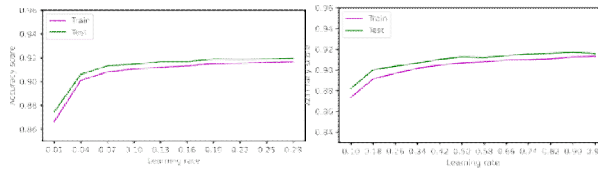


Fig.3.GNN-optimized initial travel route selection.



(a) Learning rate [0.01-0.28] (b) Learning rate [0.10-0.98]
Fig.4.Trafficcongestionpredictionaccuracy

(Fig.8) and the other in the absence of traffic congestion(Fig.9).Whenconfrontedwithtrafficcongestion,theanalysis reveals that a total distance of 4.8 kilometers is covered. During this journey, fuel consumption rates exhibit variability, ranging from [0.5-1] liters per kilometer, while carbon emission rates span the spectrum of [0.4-1.6] kilograms.These figures meticulously capture the intricate interplay of these factors, showcasing their impact on the efficiency and environmental footprint of the transportation process. This reflects the versatility and adaptability of IM-VRM model whendealingwithchallengingtrafficconditions.Incontrast, under ideal conditions, when traffic congestion isabsent, the distance traveled extends to 12 kilometers. Thefuel consumption rate in this scenario fluctuates between [0.75-2.5] liters per kilometer, while carbon emissions spantherangeoff[2-4.5]kilograms.Thesefindingsemphasize the significant adjustments in distance, fuel consumption, and carbon emissions that can be achieved when operating in favorable, congestion-free circumstances, further illustrating the capabilities of our IM-VRM model in optimizing various aspects of transportation efficiency.

Fig. 10presents a comparative analysis of green parameter values using two different methods: one with the proposed approachandtheotherwithoutit.Thisanalysisispertainto a specific journey taken by a middle-type Van passenger, encompassing five depots.

The results of this experimental case study demonstrate the noteworthy advantages of our proposed vehicle traffic managementsystem.Byincorporatingthissystem,weobserved a substantial reduction in various key metrics, namely, the distance traveled, travel time, carbon emissions, and fuel consumptionforeachdepotorhopalongtheroute.This, inturn,contributestoanoverallenhancementofgreenparam-eters,assummarizedinTableV.Thepivotalreasonfor such significant improvements in multi-depot vehicle traffic management lies in the system’s ability to accurately predict and subsequently mitigate traffic congestion. This is achieved byselectingcongestion-freeroutesafterthetravelroute GNN optimization and employing the Dijkstra algorithm to find multiple optimal routes that adhere to green parameter constraints.

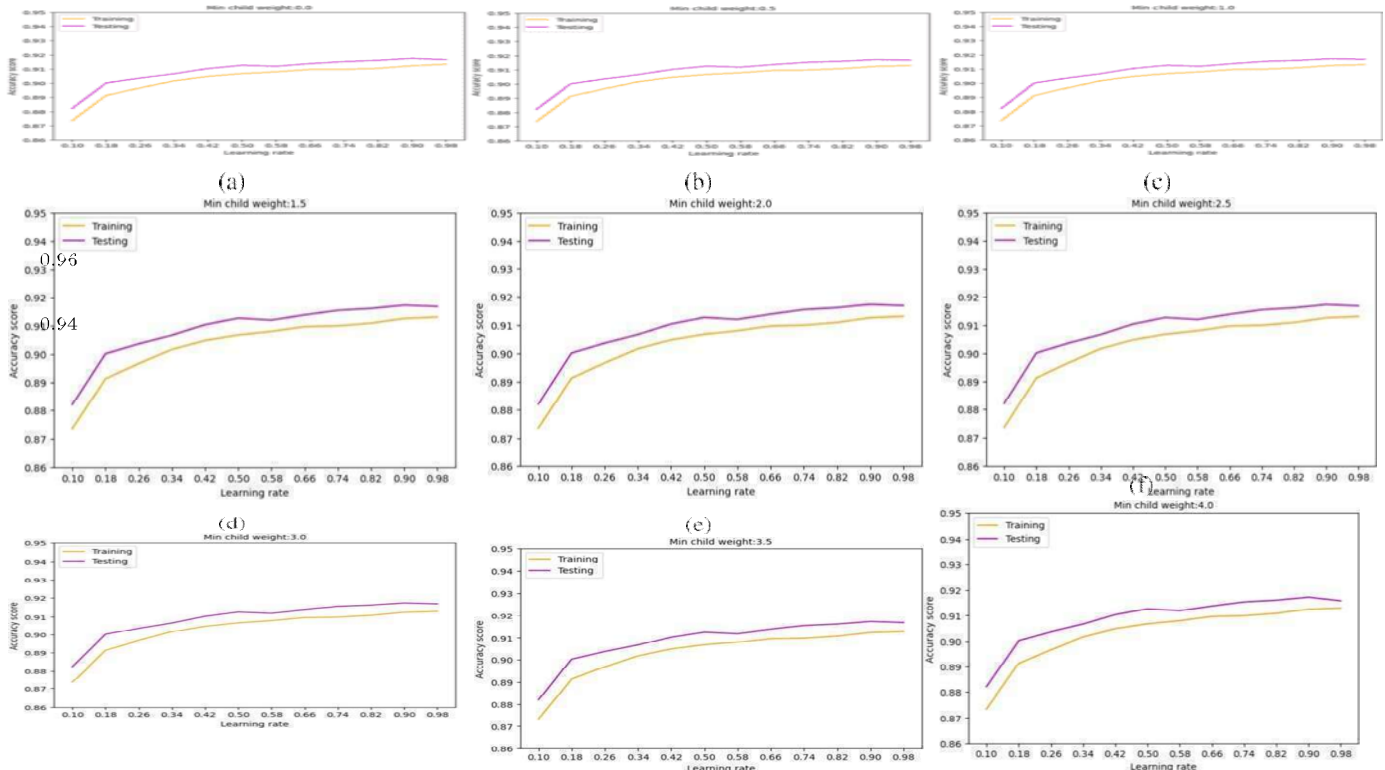


Fig.5. Trafficpredictionaccuracyovervaryingminchildweightintherange[0.0-0.4].

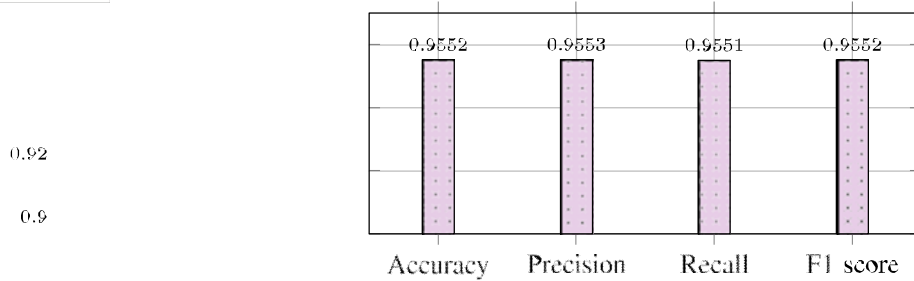


Fig.6.Trafficcongestionpredictionperformanceparameters.

TABLEV

OVERALLPERFORMANCEANALYSISOFGREENPARAMETERS

Parameter	IM-VRM	Without IM-VRM
Distance (km)	31.9	39.3
Time (minutes)	59	90
Fuel Consumption (litre)	5.2635	6.4845
CO ₂ Emission (kg)	12.1	14.94

E. Comparison

Table VI presents a comparative analysis of the proposed IM-VRM model against state-of-the-art (SoTA) approaches, including *Periodic Traffic Data Convolution-based deep Neu-ralNetwork modeling(PCNN)*[14],*LongShort-TermMemory*

TABLEVICOMPARISONWITHSOTA

Models	Approach Analysis			Loss	
	Method	Dataset			
PCNN [14]	Deep Convolution Network	Vehicle Passage Records		30.71	0.557
LSTM [15]		Traffic data from Caltrans PeMS in Oakland		46.73	0.601
GNN [29]		Traffic data form Beijing and Hangzhou city		32.71	0.302
IM-VRM	XGBoost, GNN	T-Drive trajectory dataset of Beijing city [28]		12.92	0.187

Metrics RMSE

Network (LSTM) [15], and GNN [29]. IM-VRM, leveraging XGBoostandGNNontheT-Drivedataset,achieves the best results with the lowest Loss (12.92) and RMSE (0.187), demonstrating its superior accuracy and predictive capability. This improvement stems from the effective integration of graph-based learning and optimized gradient boosting, which efficiently captures spatial-temporal traffic dynamics. IM-VRM achieves a robust balance between efficiency and precision



(a) Scaled route R1 (zoomed out) (b) Travel route R1 (zoomed in) (c) Scaled route R2 (zoomed out) (d) Travel route R2 (zoomed in)
Fig.7.Optimalrouteformulti-depottrides:R1andR2.

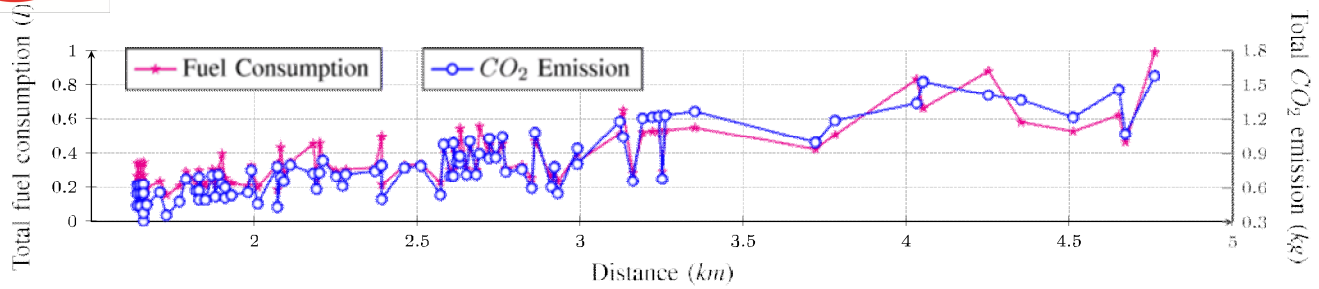


Fig.8.Evaluatedgreenparametersinpresenceoftrafficcongestion.

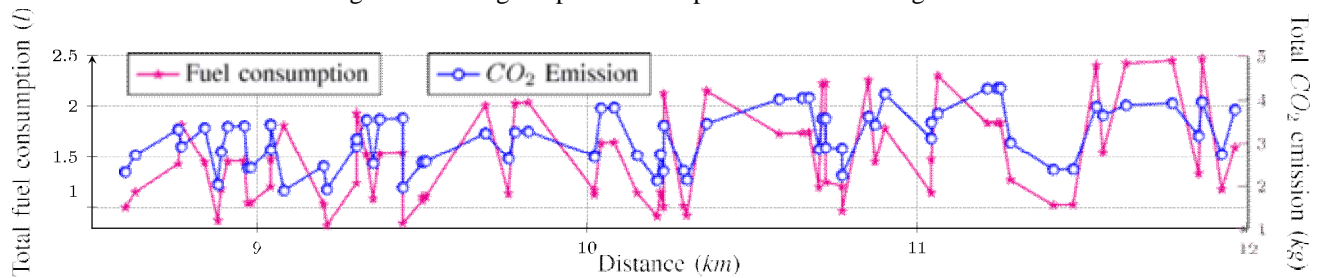


Fig.9.Evaluatedgreenparametersinabsenceoftrafficcongestion.

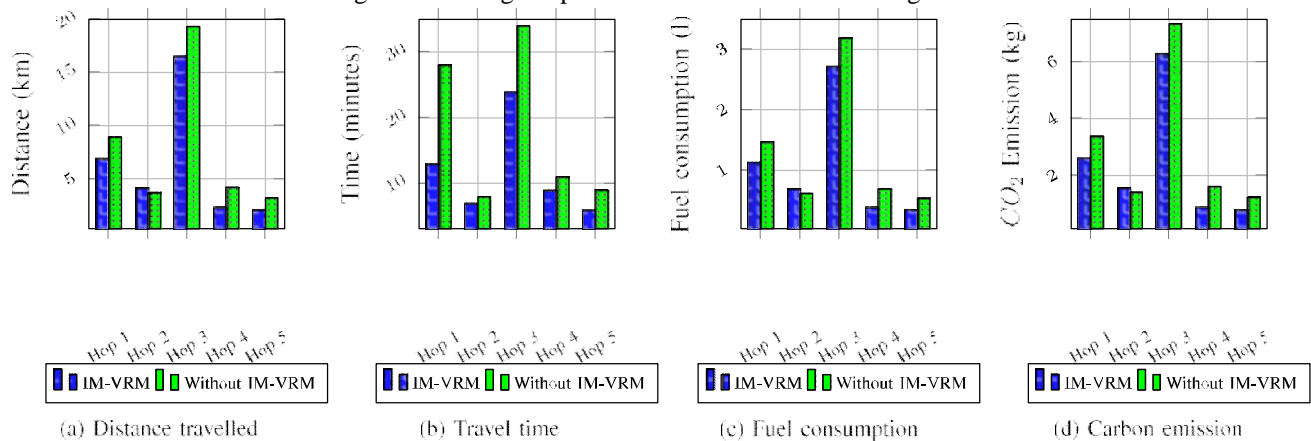


Fig.10.Greenparametersanalysis.

whileoptimizinggreenparameterswithoutcompromisingits enhanced performance.

F. OverallPerformanceAnalysisofIM-VRM

The proposed IM-VRM model exhibits exceptional performance in optimizing multi-depot delivery routes by effectively integrating GNN-based route selection, traffic congestion prediction, and green parameter constraints. Through experimental evaluation, IM-VRM achieves substantial reductions in travel distance, carbon emissions, and fuel consumption across diverse scenarios, whether in the presence or absence of traffic congestion. The model excels in predicting and mitigating congestion using accurate traffic predictions, as evidenced by an impressive 92.5% prediction accuracy at optimal learning rates [0.10-0.98]. When benchmarked against state-of-the-art methods, including PCNN, LSTM, and GNN-based approaches, IM-VRM consistently outperforms by delivering superior accuracy, evidenced by the lowest loss (12.92) and RMSE (0.187) on the T-Drive dataset. Moreover, its computational efficiency, achieved via graph-based learning and XGBoost, reflects its robustness in capturing spatial-temporal traffic dynamics. These findings affirm IM-VRM's capability to optimize transportation efficiency while promoting sustainability.

VI. CONCLUSION

This paper proposes an innovative and comprehensive approach to tackle the challenging problem of crowd delivery vehicle routing and traffic management problem. By combining GNN with greedy optimization for initial route selection and incorporating traffic congestion prediction with green parameter computation to optimize zero route guidance using the Dijkstra algorithm, we provide an efficient and effective solution.

This approach simplifies the multi-depot traffic route guidance and management problem compared to pre-viousstate-of-the-artmethods.Futureresearchwillfocus on real-time weather conditions with real-time traffic data integration and user-focused priorities to enhance practical implementation. Additionally, we will explore the scalability of the proposed approach for larger and more complex traffic networks while providing valuable insights into addressing evolving challenges in crowd delivery vehicle routing and traffic management.

REFERENCES

- [1] Q.Wang,H.Feng,H.Feng,Y.Yu,J.Li,andE.Ning,“Theimpactsof road traffic on urban air quality in jinan based GWR and remotesensing,” *Sci. Rep.*, vol. 11, no. 1, p. 15512, Jul. 2021.
- [2] J. K. Stolaroff, C. Samaras, E. R. O’Neill, A. Lubers, A. S. Mitchell,and D. Ceperley, “Energy use and life cycle greenhouse gas emissionsofdronesforcommercialpackagedelivery,”*Nat.Commun.*,vol.9,no.1, p.409,Dec.2018.
- [3] L. Wu, Y. Ci, Y. Wang, and P. Chen, “Fuel consumption at theoversaturated signalized intersection considering queue effects: A casestudy in harbin, China,” *Energy*, vol. 192, Feb. 2020, Art. no. 116654.
- [5] G. J. L. Micheli and F. Mantella, “Modelling an environmentally-extended inventory routing problem with demand uncertainty and aheterogeneous fleet under carbon control policies,” *Int. J. Prod. Econ.*,vol. 204, pp. 316–327, Oct. 2018.
- [6] H. Cheng et al., “Truck platooning reshapes greenhouse gas emissionsof the integrated vehicle-road infrastructure system,” *Nature Commun.*,vol. 14, no. 1, p. 4495, Aug. 2023.
- [7] C. Nangini et al., “A global dataset of CO₂ emissionsand ancillary data related to emissions for 343 cities,” *Sci. Data*, vol. 6, no. 1, pp. 1–29,Jan.2019.
- [8] M. Geisslinger, F. Poszler, and M. Lienkamp, “An ethical trajectory planning algorithm for autonomousvehicles,” *Nat.Mach. Intell.*,vol.5, pp.137–144,Feb.2023.
- [10] M. Maimaiti, X. Zhao, M. Jia, Y. Ru, and S. Zhu, “How we eatdetermines what we become: Opportunities and challenges brought byfood delivery industry in a changing world in China,” *Eur. J. Clin.Nutrition*, vol. 72, no. 9, pp. 1282–1286, Sep. 2018.
- [11] A. C. de Araujo and A. Etemad, “End-to-end prediction of parceldelivery time with deep learning for smart-city applications,” *IEEEInternet Things J.*, vol. 8, no. 23, pp. 17043–17056, Dec. 2021.
- [12] N.Serok,S.Havlin,andE.BlumenfeldLieberthal,“Identification,cost evaluation, and prioritization of urban traffic congestions and theirorigin,” *Sci. Rep.*, vol. 12, no. 1, p. 13026, Jul. 2022.
- [13] A. M. Avila and I. Mezić, “Data-driven analysis and forecasting ofhighway traffic dynamics,” *Nature Commun.*, vol. 11, no. 1, p. 2090,Apr.2020.
- [14] A.A.JolfaeiandM.Alinaghian,“Multi-depotvehicleroutingprob-lem with roaming delivery locations considering hard time windows:Solved by a hybrid ELS-LNS algorithm,” *Expert Syst. Appl.*, vol. 255,Dec. 2024, Art. no. 124608.
- [15] J. Du et al., “CrowDNet: Enabling a crowdsourced object deliverynetwork based on modern portfolio theory,” *IEEE Internet Things J.*,vol. 6, no. 5, pp. 9030–9041, Oct. 2019.
- [16] M.Chen,X.Yu,andY.Liu,“PCNN:Deepconvolutionalnet-works for short-term traffic congestion prediction,” *IEEE Trans. Intell.Transp. Syst.*, vol. 19, no. 11, pp. 3550–3559, Nov. 2018.
- [17] G. N. Kouziokas, “Deep bidirectional and unidirectional LSTM neuralnetworks in traffic flow forecasting from environmental factors,” in*AdvancesinMobility-as-a-ServiceSystems*,E.G.Nathanail,G.Adamos,andI.Karakikes,Eds.,Cham,Switzerland:Springer,2021,pp.171–180.
- [18] W. Joe and H. C. Lau, “Deep reinforcement learning approach to solvedynamic vehicle routing problem with stochastic customers,” in *Proc.Int. Conf. Automated Planning Scheduling*, vol. 30, 2020, pp. 394–402.
- [19] B. Wang and R. Su, “A distributed platoon control framework forconnectedautomated vehiclesinanurban trafficnetwork,” *IEEETrans.Control Netw. Syst.*, vol. 9, no. 4, pp. 1717–1730, Dec. 2022.
- [20] X.-H. Liu, M.-Y. Shan, R.-L. Zhang, and L.-H. Zhang, “Green vehicleroutingoptimizationbasedoncarbonemissionandmultiobjectivehybridquantum immune algorithm,” *Math. Problems Eng.*, vol. 2018, pp. 1–9,Jun.2018.
- [21] L.Abbatecola,M.P.Fanti,A.M.Mangini,andW.Ukovich,“Adecisionsupportapproachforpostaldeliveryandwastecollectionservices,”*IEEETrans. Autom. Sci. Eng.*, vol. 13, no. 4, pp. 1458–1470, Oct. 2016.
- [22] G.Perboli,M.Rosano,andQ.Wei,“Asimulation-optimizationapproachfor the management of the on-demand parcel delivery in sharing economy,”*IEEETrans.Intell.Transp.Syst.*,vol.23,no.8,pp.10570–10582,Aug.2022.
- [23] C.Chenetal.,“Crowddeliver:Planningcity-widepackagedeliverypathsleveragingthecrowdoftaxis,”*IEEETrans.Intell.Transp.Syst.*,vol.18,no. 6, pp. 1478–1496, Jun. 2017.
- [24] N. Kafle, B. Zou, and J. Lin, “Design and modeling of a crowdsorce-enabled system for urban parcel relay and delivery,” *Transp. Res. B,Methodol.*, vol. 99, pp. 62–82, May 2017.
- [25] H. Hong, X. Li, D. He, Y. Zhang, and M. Wang, “Crowdsourcingincentives for multi-hop urban parcel delivery network,” *IEEE Access*,vol. 7, pp. 26268–26277, 2019.
- [26] D. Saxena and A. K. Singh, “A self-healing and fault-tolerant cloud-based digital twin processing management model,” *IEEE Trans. Ind.Informat.*, early access, Feb. 26, 2025, doi: 10.1109/TII.2025.3540498.
- [27] B. Fleischmann, S. Gnutzmann, and E. Sandvoß, “Dynamic vehiclerouting based on onlinetrafficinformation,” *Transp.Sci.*,vol.38,no.4, pp.420–433,Nov. 2004.
- [29] F.An,M.Barth,J.Norbeck,andM.Ross,“Developmentofcomprehen-sivemodalemissionsmodel:Operatingunderhot-stabilizedconditions,”*Transp.Res. Record, J. Transp.Res.Board*,vol.1587,no.1,pp. 52–62,Jan.1997.
- [30] H. Li,J.Yuan,T.Lv,andX.Chang,“Thetwo-echelon time-constrainedvehicle routing problem in linehaul-delivery systems considering carbondioxideemissions,”*Transp.Res.D,Transp.Environ.*,vol.49,pp.231–245,Dec.2016.



- [31] Y. Zheng. (Aug. 2011). T-drive Trajectory Data Sample. [Online]. Available: <https://www.microsoft.com/en-us/research/publication/t-drive-trajectory-data-sample/>
- [32] M. Hou, F. Xia, H. Gao, X.Chen, and H. Chen, "Urban region profiling with spatio-temporal graph neural networks," *IEEE Trans. Computat.Social Syst.*, vol. 9, no. 6, pp. 1736–1747, Dec. 2022.



10.22214/IJRASET



45.98



IMPACT FACTOR:
7.129



IMPACT FACTOR:
7.429



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089  (24*7 Support on Whatsapp)