



OPEN ACCESS

EDITED BY

Shabana Urooj,
Princess Nourah bint Abdulrahman
University, Saudi Arabia

REVIEWED BY

Linfei Yin,
Guangxi University, China
Abdul Quadir Md,
Vellore Institute of Technology (VIT), India
Emmanouil Rigas,
Aristotle University of Thessaloniki, Greece
Navaid Z. Rizvi,
Gautam Buddha University, India

*CORRESPONDENCE

Yimin Zhou,
ym.zhou@siat.ac.cn

SPECIALTY SECTION

This article was submitted to Smart Grids, a section of the journal Frontiers in Energy Research

RECEIVED 18 July 2022

ACCEPTED 31 October 2022

PUBLISHED 13 January 2023

CITATION

Muhammad S and Zhou Y (2023), Path planning for EVs based on RA-RRT* model. *Front. Energy Res.* 10:996726. doi: 10.3389/fenrg.2022.996726

COPYRIGHT

© 2023 Muhammad and Zhou. This is an open-access article distributed under the terms of the [Creative Commons Attribution License \(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

Path planning for EVs based on RA-RRT* model

Said Muhammad and Yimin Zhou*

Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences (CAS), Shenzhen, China

The optimal path planning for EVs (electric vehicles) has gained great attention during the last decade due to the zero pollution emission characteristics and limited power capacity of EV batteries. In this paper, an optimal route search is proposed considering multiple charging stations in a dynamic urban environment, while it is still applicable when the initial available amount of the battery fails to cover a certain travel range. The TRDP (transit route design problem) and TNDP (transit node design problem) are used to search for the most feasible routes based on time and driving range via the improved route-assisted rapid random tree (RA-RRT*) algorithm. Considering the status of charge of an EV's battery during optimal routes search, three states are investigated between the destination and the aggregators: (1) bypassing the aggregators, (2) stopping over a single aggregator, and (3) stopping over multiple aggregators. During the states (2) and (3), it is required that the EVs be charged at the charging stations obtained by the RA-RRT* algorithm while approaching the destination. The proposed algorithm is tested on a random dataset under certain conditions, that is, traffic flow with congestion and assigned target locations from a given map data, with comparison experiments for efficacy verification.

KEYWORDS

path planning, heuristic model, aggregators, electric vehicles, TRDP

1 Introduction

Electric vehicles (EVs) have paved the way for green and intelligent transportation systems over the last few decades. EVs are gradually taking the place of fuel-powered vehicles due to their minimum mileage cost and environment-friendly characteristics of no carbon emissions. The main obstacle to EV's promotion is the limited battery capacity through which EVs cannot cover long distances (Zhou et al., 2020).

In recent years, EV-related energy-harvesting schemes have received a lot of attention to increase the driving range with smart route search algorithms. The related factors, that is, battery capacity, charging duration, and charging station placement, should be explicitly considered during the route planning of the EVs (El-Taweel and Farag, 2019). A reinforcement learning (RL) model is developed for EV energy management by optimal route planning (Zhang et al., 2020a), where the RL model is used to evaluate each route and approximate the energy consumption. A transit route network-based path planning

model, TRDP (transit route design problem), has been introduced to reduce the time consumption with the shortest optimal route (Yang and Jiang, 2020).

To reduce the complexity of the TRDP model, an optimization model, TNDP (transit node design problem), has been developed considering time and the xy-dimension to obtain the optimal path location (Sun et al., 2020). A significant gap is discussed between the theory and practice of the TNDP model by addressing multiple routes, nodes, and the selection of the route and node (Durán-Micco and Vansteenwegen, 2021). A multi-level network system based on different modes of transportation, that is, subway, trolleybus, and common bus, and the user constraints such as bus routes and city size has been proposed to design the transit route network in urban areas (Chavhan et al., 2021), which is applied to the Mandle dataset to reduce the travel time cost. In a study by Hellsten et al. (2021), a heuristic route network is discussed based on multi-commodity network scenarios, that is, exponentially increasing route formulation, polynomial-based path indices, and time indices. An ant colony algorithm is introduced to further reduce the complexities involved in the optimal route search task (Ghaffari et al., 2022), while a genetic algorithm (GA) can also be implied for the optimization of large-scale routes via the network data obtained from multiple cities (Bourbonnais et al., 2021). In (Miandoabchi et al., 2012), a synchronous city road network with vehicle routes has been studied via a hybrid GA and clonal selection scheme. A cloud-enabled real-time routing scheme for MoD (mobility on demand) requiring on-route charging has been developed, aiming to improve the comfort of EV users (Ammous et al., 2018). The MoD suggests that with charging delays, the total trip time for all clients will be increased in comparison to the corresponding travel lengths.

A significant amount of research has been carried out to improve the efficacy of the route search algorithms by using the traffic information from ITS (intelligent transportation systems) in terms of minimal distance and travel time cost (Tisa, 2022). The navigation systems of EVs can be updated with new technologies when the charging infrastructure is considered during route planning (Morlock et al., 2019).

An optimal route selection for EVs toward charging stations considering the traffic congestion is obtained by using Yen's algorithm with the information of transportation and distribution systems (Zhang et al., 2019). The A* algorithm is used to obtain the K-short path in order to get the K-shortest-path-joint-routing-scheduling (KSP-JRS), which can minimize the power consumption by shortest route selection while increasing the revenue via the discharge battery to the grid (Liu et al., 2020). A polynomial-time variant-based algorithm has been derived to obtain the optimal path between two target locations (Baum et al., 2020). In order to plan a simultaneous search for both velocity and path while avoiding obstacles, the A* algorithm has been implemented in multiple

dimensions, including position, velocity, and heading angle (Morsali et al., 2020). A modified and efficient random with star tree network (RRT*) model for online informative path planning (Schmid et al., 2020) has been developed to use the single search tree expansion with a continuous pattern to obtain a refined path. In a study by Qi et al. (2020), a multi-objective dynamic rapidly exploring random tree (MOD-RRT*) algorithm has been developed that can navigate the path in an unknown dynamic environment to choose an optimal node while considering the length and smoothness of the path over a short period of time. By merging the urban electrified transportation network with the power grid, an intelligent path planning method has been developed for each EV to participate in the traffic network. It can ensure the timeliness of path planning by updating the connection weights in real-time while using the Floyd algorithm to search for the shortest path in a directed weight graph (Zhou et al., 2022). An optimal route can be achieved for public buses via the Greedy and Dijkstra algorithms by labeling the available passenger pick-up and drop-off station data with unique identities (Wei et al., 2020). To achieve an optimal and free route, an extended framework of the RRT has been introduced by utilizing the way-points of a straight line while producing a collision-free path (Hu et al., 2020). Considering the EVs' charging demand and distribution network penetration, a spatial and temporal distribution prediction model has been developed with reliability assessment (Zhang et al., 2020b).

Although rapid RRT* can be used to overcome the time and space complexity of the RRT, the RRT* is not capable of achieving optimality in a finite amount of time. Thus, an RA-RRT* (route-assisted rapid random tree) model is proposed in the paper to obtain the minimized power consumption in EVs via the shortest path toward the destination, where the initial route is obtained by the RRT* algorithm as the estimation to configure the exploration. To achieve the optimal route in a short time, the RA-RRT* can achieve fast convergence by sampling. The main contributions of the paper are summarized as follows:

- A new cost function is defined in TRDP, considering both the time and EV optimal demand of the route search with a given route set. The cost function is written as a linearized mathematical modeling scheme in order to calculate the optimal route calculation with minimized time and distance.
- An RA-RRT* algorithm is proposed to generate a route set based on the demand instructions in response to the departure and destination.
- The obtained model is deployed in an undirected environment and compared with the A* and Dijkstra algorithms to verify the efficiency of fast target location detection in a wide domain area.

The remainder of the paper is organized as follows: **Section 2** describes the system model and problem formulation. The

RA-RRT* algorithm design and implementation in a dynamic environment are discussed in Section 3. Experiments and results are analyzed in Section 4. The conclusion is provided in Section 5.

2 Problem formulation

Here, multiple routes can be selected for large-scale vehicles with different types of aggregators and battery replacement centers, as depicted in Figure 1. The aggregators are divided into two categories: fast charging (FC) and slow charging (SC). The route is named EV assist route (EV-AR), which allows the users to charge or replace the battery before reaching the low limit SOC_{min} (minimum state of charge) so as to complete the current travel demand.

In Figure 1, the road network is directed by graph point (destination) $G = (CS_i, R)$, and $CS_i = \{1, 2, \dots, n\}$ represents the charging stations and R represents the set of routes. A route R is defined as a path of EVs to G and CS_i . The involved variables are listed in Table 1. The charging stations, departure points, and destination points are used to formulate the optimal route for path planning. Hence, the demand matrix can be extracted from the given set of the charging stations $M_D = \{d_{ij} | (i, j) \in CS_i\}$ and the travel time cost matrix is formulated through route search $T = \{t_{ij} | (i, j) \in R_N\}$. TRDP requires searching the paths through the objective function of route R_N over graph G . Selection of D is dependent on several factors, that is, availability of the regulators and specific charging hours.

The real-time data are collected from the Shenzhen Transportation Bureau. The TRDP model indicates its applicability for EV-AR with discrete and non-linear characteristics. The TNDP (Transit Node Design Problem) is further considered for the optimal selection of the aggregators (Weng et al., 2020) (Nikolić and Teodorović, 2014). In order to

TABLE 1 The involved parameters in the path planning.

| Notation | Unit | Definition |
|---------------|----------------------|--|
| R | 1, 2, 3, ... | Set of the EV routes |
| E_i | $i = 1, 2, 3, \dots$ | Departure point |
| T | ms | Time slot |
| SOC_{RA} | kWh | Battery level after arrival at the target location |
| CS_i | $i = 1, 2, 3, \dots$ | Charging stations |
| SOC_R | kWh | Remaining battery level at the charging station |
| E_M | km/kmh | EV's mileage |
| SOC_{max} | kWh | Fully charged battery |
| D | km | Distance |
| C_T | s | Time cost (second) |
| EV_T | h | EV's total traveling time |
| C_{Ei} | kW | Charging efficiency |
| EV_R | km | EV route |
| CS_h | h | EV charging time in hours at charging station i |
| δCS_i | $i = 1, 2, 3, \dots$ | Stop over charging station |
| S_i | $i = 1, 2, 3, \dots$ | Shared edges of the route R |
| G | | Target location (destination) |

obtain an efficient assistant route network model from the TRDP and TNDP models, the following assumptions are made:

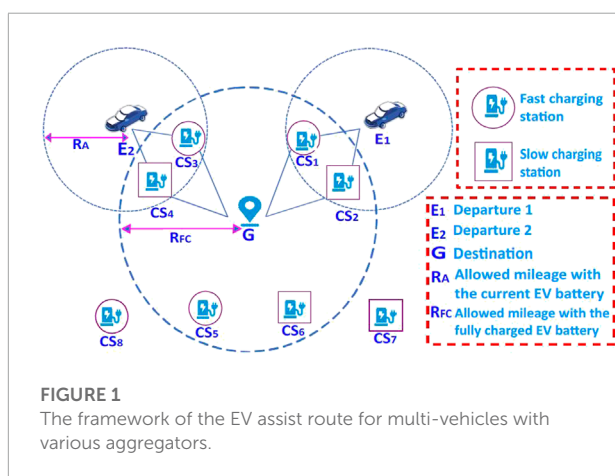
- 1) The EVs' road network is considered to be indirectly connected;
- 2) The distance between the charging stations is within the range of EVs' battery limit;
- 3) The aggregators are enough on the EVs' route to fulfill the EVs' charging demand;
- 4) When the EV user needs to travel long distances, a battery exchange scheme could be introduced to avoid more charging time;
- 5) The power supply from the G2A (Grid to Aggregator) can provide enough power to the EV users.

The efficiency of the EV-AR design has to be evaluated, and certain assumptions are made to evaluate the solution of TRDP. In order to minimize the travel distance cost, the aggregators are placed at the maximum traffic flow positions. The availability of more electric piles is considered at each charging station to avoid congestion at one charging station.

3 The RA-RRT* model for path planning

In this paper, an optimal path is designed where the EVs can easily reach the charging station, battery replacement center, and destination with minimal route selection and maximum mileage.

Let a space X be configured, where the traffic congestion is defined as X_{obs} , the free region is $X_{free} = X/X_{obs}$ and the goal G is the destination. A feasible route $x(t) \in X_{free}$ can be found via the RRT* in the T interval, starting from $r [0, T] \in R$, which begins from $X(0) = E_{start}$ to $X(T) = E_{End}$, consistent with the



given constraints of the system. During the search for the feasible routes, RRT* conserves a tree network $\tau = (E_i, G)$ of the vertices (starting points) E_i from the space X_{free} , while G represents the destination. The RRT* algorithm can be performed as follows:

3.1 Sampling

It is a random process where a route $r_{rand} = X_{free}$ is taken from the traffic congestion-free space.

3.2 Distance

It provides the route cost between target points without traffic congestion. The cost can be calculated in the form of Euclidean distance.

3.3 Nearest neighbor

Function $Nearest(\tau, r_{rand})$ is used to obtain the maximum nearby nodes between $\tau = (E_i, G)$ to r_{rand} with respect to the distance cost.

3.4 Collision check

The collision check is obtained by $Obstacle_{free}(X)$ while examining the route $x [0, T]$ if it lies in range of the free space X_{free} during the $[0, T]$ interval.

3.5 Nearby vertices

Function $Near(\tau, r_{rand}, n)$ provides the adjacent nodes in the space $\alpha(\log n/n)$ around r_{rand} where α is a constant dependent on the route plan, while n is the iteration number.

3.6 ExtractNode

In order to obtain a new node, the state function $ExtractNode(r_{initial}, r_{new}, \tau)$ adds a route r_{new} to the tree $\tau = (E_i, G)$ and connects it to the previously existing node $r_{initial}$.

3.7 Reroute

The state function $Reroute(R_{near}, r_{min}, r_{new})$ examines the routes to check whether the r_{new} is less than the older route cost.

```

1:  $\tau \leftarrow SetTree()$ ;
2:  $\tau \leftarrow ExtractNode(\Phi, r_{init}, \tau)$ ;
3: for  $i=0$  to  $i=N$  do
4: if  $i=n+A, n+2A, n+3A, \dots$  then
5:  $r_{rand} \leftarrow Sample(i, r_{CS_i})$ ;
6: else
7:  $r_{rand} \leftarrow Sample(i)$ ;
8:  $r_{nearest} \leftarrow Nearest(\tau, r_{rand})$ 
9: if  $Obstacle_{free}(x_{new})$  then
10:  $R_{near} \leftarrow Near(\tau, r_{new}, |E|)$ ;
11:  $r_{min} \leftarrow SelectParent(R_{near}, r_{nearest}, r_{new}, x_{new})$ ;
12:  $\tau \leftarrow ExtractNode(r_{min}, r_{new}, \tau)$ ;
13:  $\tau \leftarrow Reroute(\tau, R_{near}, r_{min}, r_{new})$ ;
14: if  $InitialRouteFound$  then
15:  $n \leftarrow i$ ;
16:  $(\tau, routecost) \leftarrow RouteOptimization(\tau, r_{init}, r_{goal})$ ;
17: if  $(routecost_{new} \leq routecost_{old})$ 
18:  $r_{CS_i} \leftarrow RouteOptimization(\tau, r_{init}, r_{goal})$ ;
19: return  $\tau$ 

```

Algorithm 1. $\tau = (E_i, G) \leftarrow RA-RRT^*$ Algorithm.

If the state $r_{new} > r_{previous}$, the new route is replaced by the old one due to the minimum distance cost.

The steps involved in the RA-RRT* are defined in Algorithm 1. Steps 1 to 4 and 7 to 13 are the same as those in RRT* algorithm, while the RA-RRT* algorithm is modified by the incursion of Steps 5 to 6 and 14 to 19. In the proposed RA-RRT* algorithm, the initial found route (e.g., in Step 14) returns the iteration number n which is further used for biased sampling so as to insert the time interval b (Step 4), to repeat the tree search growth during that period. The optimized route is obtained via the function $(\tau, routecost) \leftarrow RouteOptimization(\tau, r_{init}, r_{goal})$ by connecting the feasible nodes from the initial route r_{init} to the destination r_{goal} and returns the cost in Step 16. A connected topology network is depicted in Figure 2 with random route costs among different nodes. The route topology network in Figure 2A describes the distance in km between different charging stations, while the possible routes are extracted in Figure 2B. In Step 18, r_{CS_i} is the optimized routes' selection to the charging stations obtained from the function $r_{CS_i} \leftarrow RouteOptimization(\tau, r_{init}, r_{goal})$, where a new route $r_{CS_{new}}$ can be chosen over an old route $r_{CS_{old}}$ with the cost condition as $r_{CS_{new}} < r_{CS_{old}}$.

As shown in Figure 1, multiple routes toward G are taken, named EV-AR. The assist routes point the EVs toward G and different charging stations (CS_i), which play a vital role in EV energy management. While approaching G , the first task is to select the potential CS_i with the help of EV-AR along with optimal route selection.

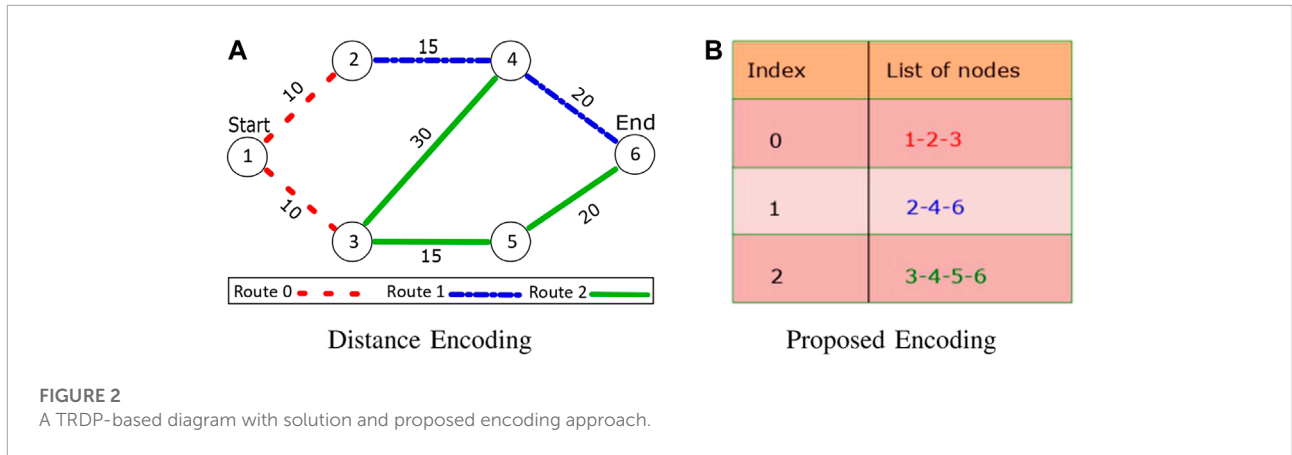


FIGURE 2 A TRDP-based diagram with solution and proposed encoding approach.

In order to obtain the EV-AR, it is required to search for the most cost-effective routes in terms of less power consumption, less distance, and maximum mileage considering traffic congestion and road infrastructure. The range R_A of the EVs can be calculated using the current battery level (SOC_R) and EVs' mileage (E_M) as follows,

$$R_A = SOC_R \times E_M. \tag{1}$$

The full range R_{FC} can be obtained with the fully charged battery level SOC_{max} and E_M ,

$$R_{FC} = SOC_{max} \times E_M. \tag{2}$$

As described earlier, the selection of the charging stations for EVs is based on their locations and types of charging. With FC aggregator, the charging time duration is only 30 min to be fully charged (Duan et al., 2020), but it requires an isolated infrastructure of power supply, which is not economically feasible.

In the proposed scheme, the charging aggregator selection is defined as the charging efficiency C_{Ei} , which is equal to the output of the electric power. The selection of the route in terms of a charging aggregator can be defined in the following scenarios:

1. Bypassing the aggregators;
2. Stopping over single aggregator;
3. Stopping over multiple aggregators

3.7.1 Bypassing the aggregators

If the EVs have enough power to reach the destination, the conventional route calculation, that is, Dijkstra (Algorithm 2), is considered to be efficient. Hence, the EVs' remaining battery level at the arrival point (SOC_{RA}) is obtained with the SOC_R at departure E_i , route distance $D(Start, End)$, and E_M ,

$$SOC_{RA} = SOC_R - D(Start, End)/E_M. \tag{3}$$

During the bypassing, the status $SOC_{RA} > 0$ is always satisfied.

3.7.2 Stopping over single aggregator

When the EVs cannot reach their destinations with the SOC_R , the users have to charge the battery on their routes. In Figure 1, the EVs' R_A level is denoted by the two small circles with two departure points E_1 and E_2 , while the R_{FC} is represented by a large circle with the center of the destination G .

The overlapped areas show that the EVs have to be recharged if $R_{FC} > R_A$. Therefore, it is necessary to search for the optimal route toward the CS_i while covering the distance D to G . The overlapped area around the EVs is distributed irregularly; hence, Euclidean distance is used to calculate D to the nearest aggregator.

The charging time cost C_T from the initial point to the target while reaching a single aggregator can be calculated as,

$$C_T(E_i, G) = \frac{\left(SOC_{max} - SOC_R + D \left(\frac{E_i, CS_1}{E_M} \right) \right)}{C_{Ei}}. \tag{4}$$

By setting E_1 as the charging point $CS_h(1, 2, 3, \dots)$, Eq. 4 can be rewritten as,

$$C_T(CS_h, CS_1) = \frac{\left(D \frac{(CS_h, CS_1)}{E_M} \right)}{C_{Ei}}. \tag{5}$$

where h is the charging time. Similarly, the total travel time cost from the starting point to the endpoint while stopping by a single aggregator can be obtained as,

$$EV_T(E_i, G) = T(E_i, CS_1) + C_T(E_i, CS_1) + T(CS_1, G). \tag{6}$$

$EV_T(E_i, G)$ and EVs battery SOC_{RA} are used as the EV-AR, and then the EVs' total routes EV_R can be calculated as follows,

$$EV_R(E_i, G) = R(E_i, \delta CS_1) + R(\delta CS_1, G). \tag{7}$$

where δCS_1 is the charging station, while R is the route between two adjacent nodes presented in Figure 2A. Similarly, EVs' total

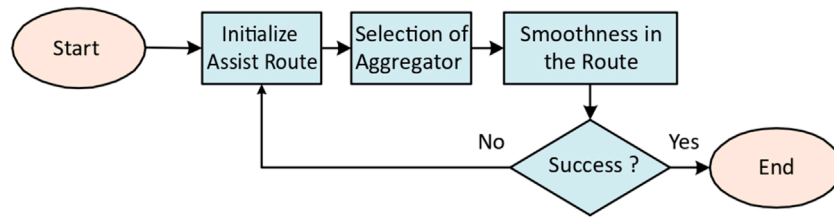


FIGURE 3
The illustration of the RA-RRT*-based route planning framework.

travel distance EV_D is calculated as,

$$EV_D(E_i, G) = D(E_i, \delta CS_1) + D(\delta CS_1, G). \quad (8)$$

It can further be modified with N number of search targets,

$$EV_D(E_i, G) = \sum_{i=1}^N D(E_i + \delta CS_1 + G). \quad (9)$$

and SOC_{RA} of EVs at the destination is calculated as,

$$SOC_{RA}(E_i, G) = SOC_{max} - D\left(\frac{\delta CS_1, G}{E_M}\right). \quad (10)$$

3.7.3 Stopping over multiple aggregators

The EVs have to be charged at more than one aggregator when traveling a long distance. The charging time cost with multiple aggregators is calculated as,

$$C(E_i, G) = C_T(E_i, CS_i) + T(\delta CS_j, CS_i) + \dots + T(CS_j, G). \quad (11)$$

Then, the route can be obtained as,

$$EV_R(E_i, G) = R(E_i, \delta CS_i) + R(\delta CS_i, \delta CS_2) + \dots + R(\delta CS_j, G). \quad (12)$$

whereas the optimal route equation can be modified as the successive product of routes set from the departure point to the destination,

$$EV_R(E_i, G) = \sum_{i=1}^N \sum_{j=1}^N R(E_i + \delta CS_j + G). \quad (13)$$

and the overall distance from the starting point to the destination is calculated as,

$$EV_D(E_i, G) = D(E_i, \delta CS_i) + D(\delta CS_i, \delta CS_2) + \dots + D(\delta CS_j, G). \quad (14)$$

To sum up, the total time taken by the EVs from the starting point to the destination is expressed as,

$$EV_T(E_i, G) = T(E_i, \delta CS_i) + C_T(E_i, \delta CS_1) + T(\delta CS_1, \delta CS_2) + C_T(\delta CS_1, \delta CS_2) + \dots + T(\delta CS_{j-1}, \delta CS_j) + C_T(\delta CS_{j-1}, \delta CS_j) + T(\delta CS_j, G). \quad (15)$$

Similarly, the remaining battery level at the destination can be obtained from the statistics of SOC_{max} and overall covered distance,

$$SOC_{RA}(E_i, G) = SOC_{max} - \sum_{j=1}^N D\left(\frac{\delta CS_j, G}{E_M}\right). \quad (16)$$

3.7.4 Assistive route analysis

The developed RA-RRT* model is implied on a given set of data collected from the Public Transport Bureau of Shenzhen, Guangdong Province, China, including the longitudinal angles, latitudinal angles, and the public transport terminal locations. The retrieved routes by RA-RRT* can be reduced in length with the dynamic constraints on its trajectory. The entire framework is depicted in **Figure 3**, which is implemented as follows:

A suitable aggregator is selected considering the route and battery status details, that is, R_A, R_{FC} . In order to obtain the smooth route, an efficient route formulation strategy is jointly used with the obtained route details (**Eqs 1-16**). The routes are plotted based on the provided details of the route transfer mechanism among different traffic terminals using python gmpplot function. Furthermore, the latitudinal and longitudinal angular data are converted into distance by using

```

1: Function Dijkstra {Ei,G}
2: Nodes initialization
3: Data (Distance between Nodes) initialization
4: for each node Ei in G
5:     distance[V] ← infinite
6:     previous[V] ← NULL
7:     If Ei != G, add Ei to the priority
        route of the map
8: distance[G] ← 0
9: while route R is undirected
10:    Ei ← extract minimum from the route
11:    for each unvisited neighbor from V of
        Ei
12:        tempDistance ← distance[Ei]
13:            +
        edge-weight(Ei, V)
14:        if tempDistance < distance[V]
15:            distance[V] ← tempDistance
16:            previous[V] ← G
17: return distance[], previous[]
    
```

Algorithm 2. Shortest path search via Dijkstra algorithm.

```

1: Import the gmap from the plotting
2: Initialize the list of latitude and
    longitude
3: #Google Map location selection
4: gmap3 = gmap.GoogleMapPlotter(initial,
    final, size)
5: #Scatter point creation
6: gmap3.scatter(latitude_list, longitude_
    list, '#FF000',
7:             size = , marker = )
8: Draw line b/w the given co-ordinates
9: #Plotting the lines between charging stations
10: gmap3.plot(latitude_list, longitude_list,
    'color',
11:            edge_width = 3.0)
12: #Insert the gmap apikey
13: gmap.apikey = '-----'
14: #Creating .html file
15: gmap3.draw('` File location to create
    .html file ")
    
```

Algorithm 4. Target area mapping.

```

1: MAP = (Ei, CSi)
2: np ← import numpy
3: lati ← import latitudinal angular data
4: loni ← import longitudinal angular data
5: Function distance{latang index, lonang index
    }
6: #From angular data to radian conversion
7:     latrad ← np.deg2rad(latang)
8:     lonrad ← np.deg2rad(lonang)
9: #by using the Haversine formula
10:    distance ← difference of the
        latrad and lonrad
11:
12: Display distance in Km
    
```

Algorithm 3. Distance mapping among different charging stations CS_i.

the Haversine Formula in Algorithm 3 as follows,

$$\begin{aligned}
 a &= np.\sin\left(\frac{d_{lat}}{2}\right)^2 + np.\cos(\ell_{i-1}) \times np.\cos(\ell_i) \\
 &\quad \times np.\sin\left(\frac{d_{lon}}{2}\right)^2 \\
 c &= 2 \times np.\arcsin(np.\sqrt{a}) \\
 d &= c \times r
 \end{aligned}$$

where d_{lat} is the error between the two consecutive latitudinal angles lat_i , while d_{lon} is the error between the two consecutive longitudinal angles lon_i and np is the python *numpy* library which

is a source for multiple functions, that is, N-dimensional array. c and d are constant variables, while r is the radius of the Earth, equal to 6,371 km.

4 Experiment and result analysis

In this section, experiments are performed to verify the efficacy of the proposed RA-RRT* model for EV optimal path planning. The experiments are conducted on a Windows operating system with an Intel Core i7 with a RAM of 16 GHz and 8 GHz processor. Simulation experiments are carried out by using the software tool Python (Jupyter notebook and PyCharm environment). GMPLOT, a Google map plot function built into the Python library, is used to obtain the locations on the map with latitudinal and longitudinal angles retrieved from different target locations. In Figure 4A, the target locations are identified using a heat map function, while the routes between the target locations are plotted in Figure 4B. The route distances between different public transport terminals can be calculated via the heuristic (Dijkstra) algorithm.

Two scenarios have been considered to validate the RA-RRT* model. During the 1st scenario, the RA-RRT* model is used for the optimal route selection between the target locations, that is, start point, endpoint, and charging station CS_i, while the algorithm is implemented for traffic congestion avoidance in the 2nd scenario.

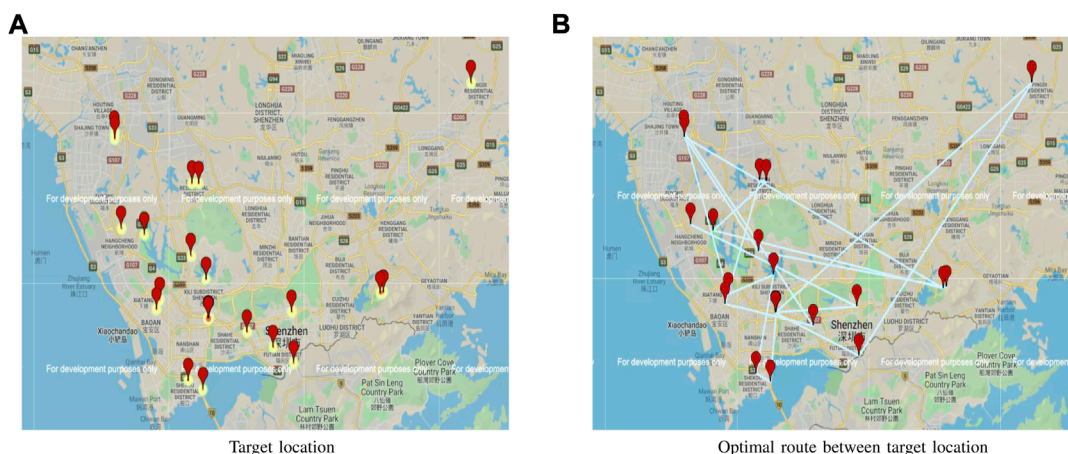


FIGURE 4 A route network among different charging stations in Shenzhen (2021).

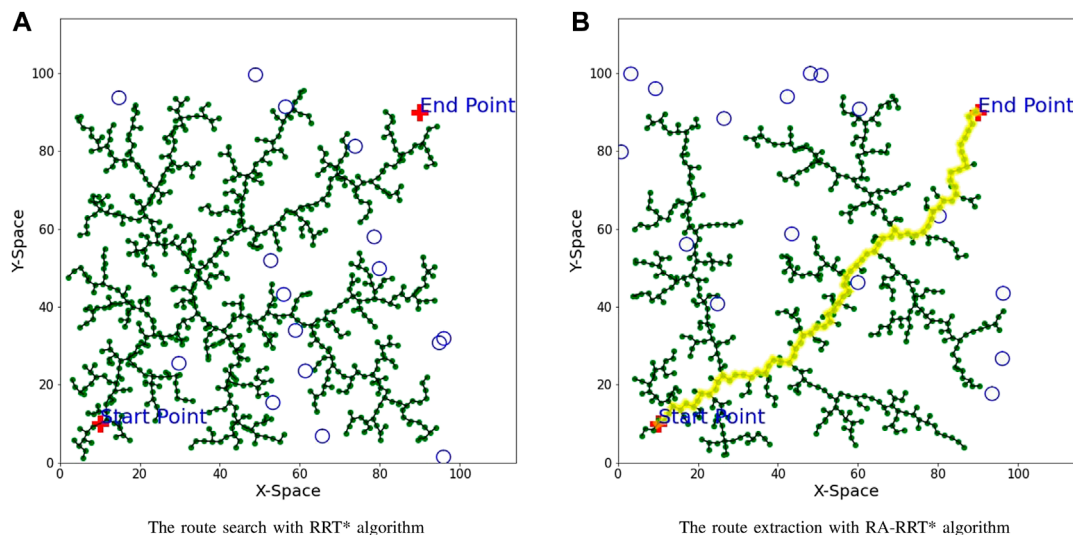


FIGURE 5 Optimal path achievement with the nearest charging stations with RRT* and RA-RRT* algorithms.

4.1 Optimal route selection

The developed RA-RRT* algorithm is used on the target locations obtained by the Haversine formula (Mahariba et al., 2022) in Figure 4, from a given set of data. The average mileage per kilowatt-hour (kWh) is taken as 6 km, while the full battery capacity is taken as 24 kWh.

The known initial SoC_{ini} of the EV is used to calculate R_A and R_{FC} . The target locations are presented with the start points (highlighted in Figure 5) and final point in an undirected and

non-convex environment in Figure 5. Based on the given target locations, the distance, point of aggregators, and route search time are calculated and listed in Table 2.

In Figure 5A, the starting point and the end point are taken at axis (10,10) and (90,90), respectively. The charging stations are denoted as circles in the free search area. Figure 5B shows the results of the optimal routes between two points via the proposed RA-RRT* algorithm. While approaching the target location with the shortest route, it has considered passing across the route if it has enough aggregators. The obtained route

TABLE 2 Search time results for routes' and aggregators' selections with the proposed RA-RRT* algorithm.

| No. | Start point (departure position) | Endpoint (destination position) | SOC _R (kWh) | R _A (km) | Travel distance (km) | Travel time (m) | # Of δCS _i | POI search time (s) | Route search time (s) | Total search time (s) |
|-----|----------------------------------|---------------------------------|------------------------|---------------------|----------------------|-----------------|-----------------------|---------------------|-----------------------|-----------------------|
| 1 | Fukuda Ward | Bao'an District | 6 | 36 | 50.321 | 87 | 1 | 3.6 | 4.5 | 8.1 |
| 2 | Longgang District | Nanshan District | 12 | 72 | 58.998 | 109 | 0 | 3.8 | 4.5 | 8.3 |
| 3 | Nanshan District | Fukuda Ward | 8 | 48 | 56.458 | 99 | 0 | 3.5 | 5.1 | 8.6 |
| 4 | Nanshan District | Nanshan District | 9 | 54 | 78.325 | 150 | 1 | 4.8 | 5.4 | 10.2 |
| 5 | Nanshan District | Nanshan District | 15 | 90 | 120.251 | 220 | 1 | 5.8 | 9.8 | 15.6 |
| 6 | Bao'an District | Bao'an District | 12 | 72 | 160.546 | 300 | 2 | 6.8 | 15.6 | 22.4 |
| 7 | Bao'an District | Fukuda Ward | 20 | 120 | 150.234 | 280 | 1 | 7.9 | 11.6 | 19.5 |
| 8 | Bao'an District | Bao'an District | 18 | 108 | 98.365 | 180 | 0 | 6.5 | 10.8 | 17.3 |
| 9 | Bao'an District | Bao'an District | 16 | 96 | 88.991 | 167 | 0 | 4.8 | 12.5 | 17.3 |
| 10 | Bao'an District | Bao'an District | 7 | 42 | 48.450 | 90 | 0 | 5.8 | 7.5 | 13.3 |
| 11 | Luohu District | Fukuda Ward | 22 | 132 | 158.453 | 286 | 2 | 6.5 | 15.5 | 22 |
| 12 | Bao'an District | Bao'an District | 10 | 60 | 78.324 | 140 | 0 | 4.6 | 5.5 | 10.1 |
| 13 | Nanshan District | Nanshan District | 11 | 66 | 59.451 | 111 | 0 | 3.9 | 6.5 | 10.4 |
| 14 | Nanshan District | Bright Zone | 13 | 78 | 150.477 | 282 | 1 | 7.8 | 13.5 | 21.3 |
| 15 | Bao'an District | Bright Zone | 11 | 66 | 170.563 | 320 | 2 | 7.5 | 16.5 | 24 |

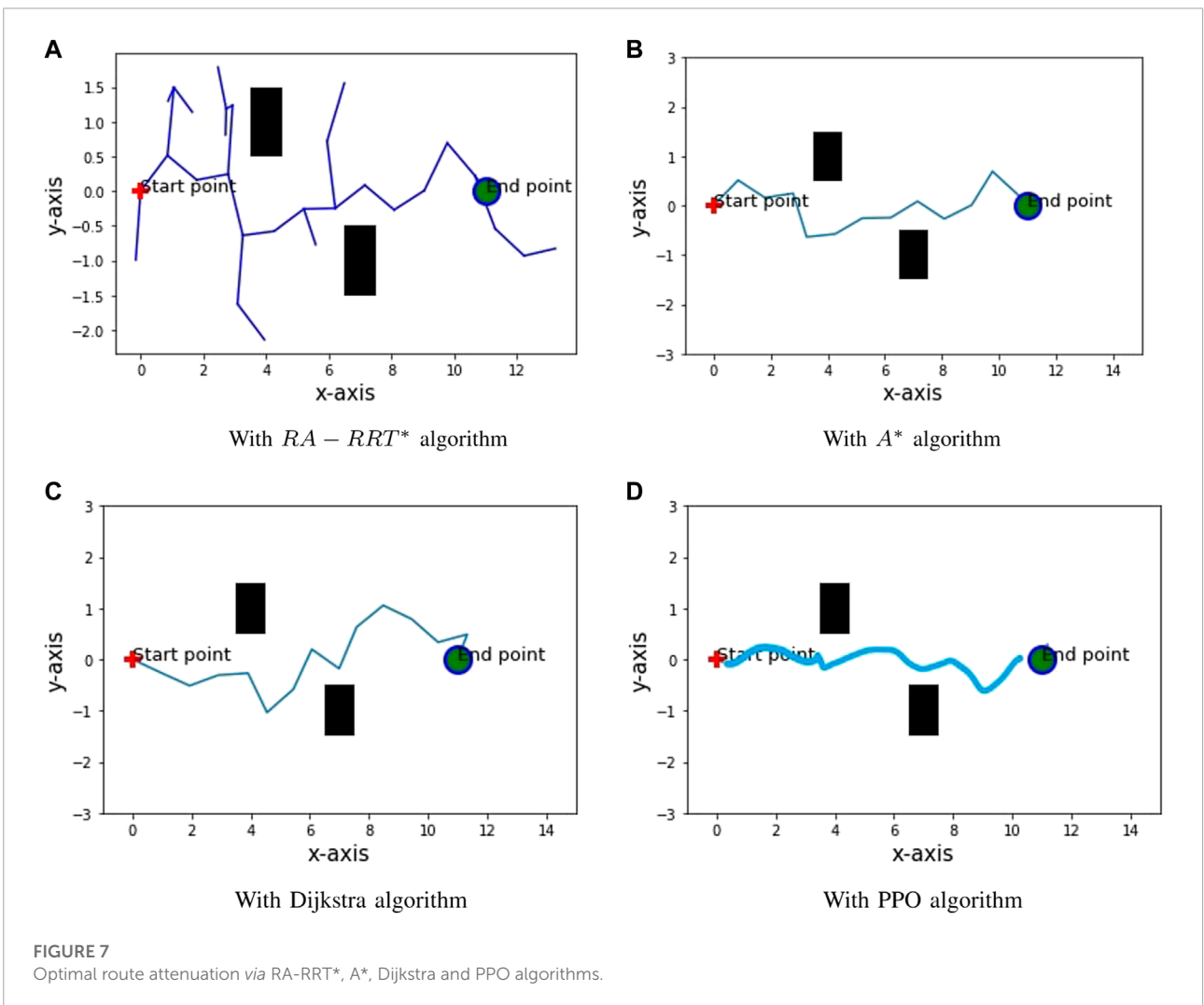
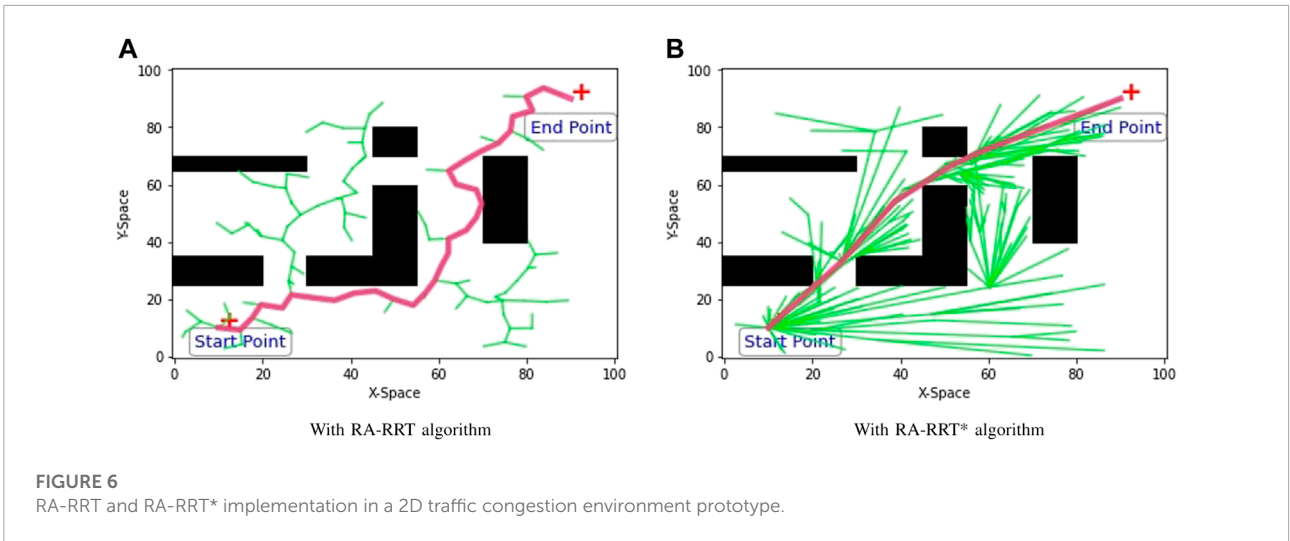


TABLE 3 The performance comparison of the three algorithms.

| No. | Search Algorithms | No. of Iterations | Time cost (s) | Distance (km) |
|-----|-------------------|-------------------|---------------|---------------|
| 1 | RA-RRT* | 16 | 4.82 | 50 |
| 2 | A* | 55 | 7 | 56 |
| 3 | Dijkstra | 67 | 5 | 58 |
| 4 | PPO | 70 | 9 | 59 |

is highlighted with a yellow color, which is smooth between the start point, charging station allocation, and destination. In order to avoid the charging shortage, an early warning model about insufficient battery levels can be developed by using the appropriate energy consumption factor (Wang et al., 2021).

4.2 Bypassing the traffic congestion

The proposed RA-RRT* algorithm is further elucidated in a 2D environment considering the traffic congestion to be formed as blocks X_{obs} with the indirect connection between the starting point and the destination, shown in Figure 6A.

In Figure 6A, the departure point is taken at (10,10) with map location at north-south, while the destination point is taken at (90,90), the east-west positions, respectively. It is a non-smooth path search in the free space $X_{free} = X/X_{obs}$; hence, the RA-RRT* model is further compared with the RRT* by increasing the number of search samplings in the free space to find an initial route immediately. The RA-RRT*-based graph is achieved in Figure 6B, illustrating the improved results in terms of route smoothness and length.

4.3 The comparison among path planning algorithms

The performance of the proposed RA-RRT* is compared with both A* and Dijkstra algorithm in Figure 7. The obtained results in Table 3 can demonstrate that the RA-RRT* is faster compared to both A* and Dijkstra algorithms in the search for the target location. In order to retrieve the optimal route toward the target location, RA-RRT* compiled 16 iterations, while A* and Dijkstra took 55 and 67 compilation iterations, respectively. The algorithms are evaluated in Table 3 based on their iteration time costs. During charging station searching and following the optimal path towards the destination, the wide-spreading search property of the RA-RRT* algorithm makes it more suitable for multiple targets searching simultaneously.

Furthermore, a deep reinforcement learning algorithm, the Proximal Policy Optimization algorithm (PPO), is used for path

TABLE 4 RA-RRT* comparison with different route search algorithms.

| Model | Speed | Time |
|----------|---------|------|
| RA-RRT* | Fastest | 4.82 |
| A* | Fast | 7 |
| Dijkstra | Fast | 5 |
| PPO | Fast | 8 |

planning comparison with the RA-RRT* model in an obstacle environment. The PPO algorithm (Sadhukhan and Rastko, 2022) performs on an iteration basis to search for the target location with more time consumption for the target location, while RA-RRT* can explore in all directions due to its tree search property with less time consumption. The PPO algorithm is carried out in a Jupyter notebook on a Windows operating system with an Intel Core i7 with a RAM of 16 GHz and 8 GHz processor.

In Table 4, a metric evaluation of the RA-RRT* algorithm with the A*, RRT*, and Dijkstra algorithms regarding speed and accuracy is shown. It is found that the A* algorithm is more efficient in a directional and known environment, whereas the Dijkstra and RRT* algorithms are more feasible in an undirected environment (Erokhin et al., 2018). Based on the experimental results, the proposed RA-RRT* is proven to be more efficient due to the requirement to simultaneously search for the charging stations and the minimum optimal path in an undirected environment because of the tree structure.

5 Conclusion

In this paper, a route search-based problem has been carried out with the proposed RA-RRT* algorithm via an assistive route heuristic model, achieving fast route search capability for multiple targets in a wide range area. Different scenarios are considered with stops at a single aggregator, multiple aggregators, or for battery replacement. A comparative analysis of the proposed model is taken with the A*, Dijkstra, and PPO algorithms, where the proposed RA-RRT* model is more feasible for multi-target fast route search results in an undirected environment. Further research will apply the proposed RA-RRT* algorithm for real-time path planning in autonomous urban driving scenarios.

Data availability statement

The raw data supporting the conclusions of this article will be made available by the authors, without undue reservation.

Author contributions

SM: investigation, methodology, simulation analysis, and writing—original draft. YZ: supervision, resources, project administration, and writing—review and editing.

Funding

This work was supported in part by the National Natural Science Foundation of China Ref. U1913201, and Ref. 61973296, in part by the Shenzhen Science and Technology Innovation Commission Project Grant Ref. JCYJ20170818153635759, Ref. JCYJ20200109114839874 and Ref. JCYJ20210324101215039.

References

- Ammous, M., Belakaria, S., Sorour, S., and Abdel-Rahim, A. (2018). Optimal cloud-based routing with in-route charging of mobility-on-demand electric vehicles. *IEEE Trans. Intell. Transp. Syst.* 20 (7), 2510–2522. doi:10.1109/tits.2018.2867519
- Baum, M., Dibbelt, J., Pajor, T., Sauer, J., Wagner, D., and Zundorf, T. (2020). Energy-optimal routes for battery electric vehicles. *Algorithmica* 82 (5), 1490–1546. doi:10.1007/s00453-019-00655-9
- Bourbonnais, P. L., Morency, C., Trépanier, M., and Martel-Poliquin, E. (2021). Transit network design using a genetic algorithm with integrated road network and disaggregated O–D demand data. *Transportation* 48 (1), 95–130. doi:10.1007/s11116-019-10047-1
- Chavhan, S., Gupta, D., Nagaraju, C., Khanna, A., and Rodrigues, J. P. C. (2021). An efficient context-Aware vehicle incidents route service management for intelligent transport system. *IEEE Syst. J.* 16 (1), 487–498. doi:10.1109/jsyst.2021.3066776
- Duan, X., Hu, Z., and Song, Y. (2020). Bidding strategies in energy and reserve markets for an aggregator of multiple EV fast-charging stations with battery storage. *IEEE Trans. Intell. Transp. Syst.* 22 (1), 471–482. doi:10.1109/tits.2020.3019608
- Durán-Micco, J., and Vansteenwegen, P. (2021). A survey on the transit network design and frequency setting problem. *Public Transp.* 14 (1), 155–190. doi:10.1007/s12469-021-00284-y
- El-Taweel, N. A., and Farag, H. (2019). Integrated utility-transit model for optimal configuration of battery electric bus systems. *IEEE Syst. J.* 14 (1), 738–748. doi:10.1109/jsyst.2019.2926460
- Erokhin, A., Erokhin, V., and Sotnikov, S. (2018). Optimal multi-robot path finding algorithm based on A*. *Proc. Comput. Methods Syst. Softw.* 860 (1), 172–182. doi:10.1007/978-3-030-00184-1_16
- Ghaffari, A., Mesbah, M., Khodaii, A., and MirHassani, S. A. (2022). Risk-based formulation of the transit priority network design. *IEEE Trans. Intell. Transp. Syst.* 23 (7), 8895–8905. doi:10.1109/tits.2021.3087525
- Hellsten, E., Koza, D. F., Contreras, I., Cordeau, J. F., and Pisinger, D. (2021). The transit time constrained fixed charge multi-commodity network design problem. *Comput. Oper. Res.* 136 (1), 105511–105536. doi:10.1016/j.cor.2021.105511
- Hu, B., Cao, Z., and Zhou, M. (2020). An efficient RRT-based framework for planning short and smooth wheeled robot motion under kinodynamic constraints. *IEEE Trans. Ind. Electron.* 68 (4), 3292–3302. doi:10.1109/tie.2020.2978701
- Liu, P., Wang, C., Hu, J., Fu, T., Cheng, N., Zhang, N., et al. (2020). Joint route selection and charging discharging scheduling of EVs in V2G energy network. *IEEE Trans. Veh. Technol.* 69 (10), 10630–10641. doi:10.1109/tvt.2020.3018114
- Mahariba, A. J., Uthra, R. A., and Brunet, R. G. (2022). Estimation of shortest route with minimum travel time using GIS and MSSTT algorithm. *Adv. Constr. Manag.* 191 (1), 565–579. doi:10.1007/978-981-16-5839-6_48
- Miandoabchi, E., Farahani, R. Z., Dullaert, W., and Szeto, W. Y. (2012). Hybrid evolutionary metaheuristics for concurrent multi-objective design of urban road and public transit networks. *Netw. Spat. Econ.* 12 (3), 441–480. doi:10.1007/s11067-011-9163-x
- Morlock, F., Rolle, B., Bauer, M., and Sawodny, O. (2019). Time optimal routing of electric vehicles under consideration of available charging infrastructure and a detailed consumption model. *IEEE Trans. Intell. Transp. Syst.* 21 (12), 5123–5135. doi:10.1109/tits.2019.2949053
- Morsali, M., Frisk, E., and Åslund, J. (2020). Spatio-temporal planning in multi-vehicle scenarios for autonomous vehicle using support vector machines. *IEEE Trans. Intell. Veh.* 6 (4), 611–621. doi:10.1109/tiv.2020.3042087
- Nikolić, M., and Teodorović, D. (2014). A simultaneous transit network design and frequency setting: Computing with bees. *Expert Syst. Appl.* 41 (16), 7200–7209. doi:10.1016/j.eswa.2014.05.034
- Qi, J., Yang, H., and Sun, H. (2020). MOD-RRT*: A sampling-based algorithm for robot path planning in dynamic environment. *IEEE Trans. Ind. Electron.* 68 (8), 7244–7251. doi:10.1109/tie.2020.2998740
- Sadhukhan, P., and Rastko, R. (2022). Selmic. “Proximal policy optimization for formation navigation and obstacle avoidance. *Int. J. Intelligent Robotics Appl.* 65 (1), 1–14. doi:10.1007/s41315-022-00245-z
- Schmid, L., Pantic, M., Khanna, R., Ott, L., Siegwart, R., and Nieto, J. (2020). An efficient sampling-based method for online informative path planning in unknown environments. *IEEE Robot. Autom. Lett.* 5 (2), 1500–1507. doi:10.1109/lra.2020.2969191
- Sun, G., Tian, Z., Liu, R., Jing, Y., and Ma, Y. (2020). Research on coordination and optimization of order allocation and delivery route planning in take-out system. *Math. Problems Eng.* 20 (20), 1–16. doi:10.1155/2020/7248492
- Tisa (2022). Tpeg. Available at: “<https://tisa.org/>”.
- Wang, Z., Song, C., Zhang, L., Zhao, Y., Liu, P., and Dorrell, D. G. (2021). A data-driven method for battery charging capacity abnormality diagnosis in electric vehicle applications. *IEEE Trans. Transp. Electrific.* 8 (1), 990–999. doi:10.1109/tte.2021.3117841
- Wei, M., Liu, T., Sun, B., and Jing, B. (2020). Optimal integrated model for feeder transit route design and frequency-setting problem with stop selection. *J. Adv. Transp.* 20 (20), 1–12. doi:10.1155/2020/6517248
- Weng, D., Zheng, C., Deng, Z., Ma, M., Bao, J., Zheng, Y., et al. (2020). Towards better bus networks: A visual analytics approach. *IEEE Trans. Vis. Comput. Graph.* 27 (2), 817–827. doi:10.1109/tvcg.2020.3030458
- Yang, J., and Jiang, Y. (2020). Application of modified NSGA-II to the transit network design problem. *J. Adv. Transp.* 20 (20), 1–24. doi:10.1155/2020/3753601

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Zhang, C., Liu, Y., Wu, F., Tang, B., and Fan, W. (2020). Effective charging planning based on deep reinforcement learning for electric vehicles. *IEEE Trans. Intell. Transp. Syst.* 22 (1), 542–554. doi:10.1109/tits.2020.3002271

Zhang, Q., Wu, K., and Shi, Y. (2020). Route planning and power management for PHEVs with reinforcement learning. *IEEE Trans. Veh. Technol.* 69 (5), 4751–4762. doi:10.1109/tvt.2020.2979623

Zhang, X., Li, P., Hu, J., Liu, M., Wang, G., Qiu, J., et al. (2019). Yen's algorithm-based charging facility planning considering congestion in coupled

transportation and power systems. *IEEE Trans. Transp. Electrific.* 5 (4), 1134–1144. doi:10.1109/tte.2019.2959716

Zhou, X., Zou, S., Wang, P., and Ma, Z. (2020). ADMM-based coordination of electric vehicles in constrained distribution networks considering fast charging and degradation. *IEEE Trans. Intell. Transp. Syst.* 22 (1), 565–578. doi:10.1109/tits.2020.3015122

Zhou, Z., Liu, Z., Su, H., and Zhang, L. (2022). Intelligent path planning strategy for electric vehicles combined with urban electrified transportation network and power grid. *IEEE Syst. J.* 16 (2), 2437–2447. doi:10.1109/jsyst.2021.3075088