Elektrikli Araç Rotalama Problemi için Uyarlanabilir Geniş Komşuluk Arama Yaklaşımı

Adaptive Large Neighborhood Search Approach for Electric Vehicle Routing Problem

Yasin Ünal[1], İnci Sarıçiçek[1,2], Sinem Bozkurt Keser[1,3], Ahmet Yazıcı[1,3]

[1]Center of Intelligent Systems Applications Research (CISAR)
Eskişehir Osmangazi Üniversitesi, Eskişehir
yasin0.unal@gmail.com

[2]Department of Industrial Engineering
Eskişehir Osmangazi Üniversitesi, Eskişehir
incid@ogu.edu.tr

[3]Department of Computer Engineering
Eskişehir Osmangazi Üniversitesi, Eskişehir
sbozkurt@ogu.edu.tr
ayazici@ogu.edu.tr

Özetçe

Son yıllarda karbon emisyon hedeflerini dikkate alan Elektrikli Araç Rotalama Problemleri yoğun ilgi görmektedir. Özellikle lojistik ve taşımacılıkta yakıt tüketiminin azaltılması ve karbon salınımının düşürülmesi amaçlanmaktadır. Bu bakımdan son yıllarda elektrikli araç kullanımı artmaktadır. Lojistikte elektrikli araç kullanımı beraberinde batarya kapasitesi ve şarj süresi gibi konuları dikkate almayı gerektirmektedir. Elektrikli araç rotalama problemleri bu doğrultuda NP-Zor problemlerdir. Bu nedenle kısa sürede kaliteli çözümler üreten meta-sezgisel algoritmalara başvurulmaktadır. Bu çalışmada Kapasiteli Elektrikli Araçlarda Rotalama Problemi (CEVRP) için hibrit Uyarlanabilir Geniş Komşuluk Arama (ALNS) algoritması kullanılmıştır. ALNS her adımda yeni çözümleri operatörleri ile üretip, bu çözümleri değerlendiren ve başarılı olan operatörlerin seçilme olasılığının arttırıldığı uyarlanabilir bir algoritma olması nedeni ile seçilmiştir. Ayrıca, çözüm uzayını daha hızlı taramak için yerel arama yöntemleri algoritmaya dâhil edilmiştir. Üç farklı yöntem oluşturularak ESOGÜ kampüs haritasından oluşturulan müşteri talep noktaları için sonuçlar incelenmiştir. Bu yöntemlerin sonuçlara olan etkisi analiz edilmiş ve yerel arama yöntemlerinin rastgele seçilen ve talep noktası fazla olan problemlerde iyi sonuçlar elde ettiği gösterilmiştir.

Abstract

In recent years, Electric Vehicle Routing Problems considering carbon emission targets has received extensive attention. Particularly in logistics and transportation, the aim is to reduce fuel consumption and lower carbon emissions. In this regard, the use of electric vehicles has increased in recent years. Using electric vehicles in logistics necessitates considerations such as battery capacity and charging times. Electric vehicle routing problems are NP-hard problems in this context. Therefore, metaheuristic algorithms that can produce high-quality solutions faster are employed. In this study, the Hybrid Adaptive Large Neighborhood Search (ALNS) algorithm is used for the Capacitated Electric Vehicle Routing Problem (CEVRP). ALNS is chosen for its adaptive nature, generating new solutions at each step using operators, evaluating these solutions, and increasing the likelihood of selecting successful operators. Additionally, local search methods are integrated into the algorithm to explore the solution space faster. Results show that local search methods are included in the algorithm to scan the solution space faster. Three methods are compared to obtain effective scanning of the solution space.in the ESOGU campus map. The effects of these methods on the results are analysed, and it is shown that local search methods achieve good results for randomly selected problems with more demand points.

1. Introduction

Carbon emissions from the transportation industry account for 15% of global carbon emissions, and its average annual growth rate has been above 5% in past years [1]. With the proliferation of electric vehicles, energy efficiency and sustainability concepts have emerged. The increasing use of electric vehicles in logistics and transportation is critical in improving efficiency and reducing carbon emissions [2]. Capacitated Electric Vehicle Routing Problem (CEVRP) is a delivery problem where the capacitated vehicle should visit customers or charging stations within a route, starting and ending at a depot. Electric vehicles bring challenges such as battery capacity and charging, leading to high computational costs in solving routing problems. Metaheuristic algorithms are widely used in this field because methods that provide exact solutions cannot provide solutions in a short time for large-scale problems. ALNS is one of the methods recommended in the literature for solving EVRPs. In the literature, the study by Ropke and Pisinger (2006) is the first to introduce the ALNS algorithm for the Vehicle Routing Problem with Time Windows, demonstrating relatively better results [3]. Demir et al. (2014) focused on energy consumption and carbon emissions, examining the performance impacts of different charging strategies and approaches [4]. Schneider et al. (2014) compared metaheuristic methods in their study, highlighting the high performance of the ALNS algorithm [5]. Keskin and Çatay (2016) addressed the time window problem for partial charging strategies, updating weights in each iteration with the ALNS algorithm. [6]. Erdelić and Carić (2019) expanded the search space using local search methods within the ALNS framework [7, 8]. Erdem and Koç (2019) applied the ALNS method to a different context, the home healthcare problem for nurses and patients, working with three charging strategies and proposed operators [9]. Bac and Erdem (2021) addressed the home healthcare problem during the COVID-19 pandemic using local search methods and ALNS. They studied using private and public charging stations based on map data from Samsun, showing that private charging stations could be used despite the cost [10]. Mara et al. (2022) created a review that included studies in the literature on ALNS and systematically examined different methods in this review. [11].

This study uses the Hybrid Adaptive Large Neighborhood Search (ALNS) algorithm to solve CEVRP for minimizing the total distance. The problem data is generated using the ESOGU campus map. The fact that the study is conducted with a real environment data has contributed to the literature in this field. Furthermore, the full charging strategy is considered, and operators that explore the solution space faster are included to improve the solution quality. With these operators, the ALNS algorithm is extended with swap and local search methods, and their performances are compared. The route obtained from the algorithm is repaired considering the charging capacity. Charging stations are added to the required locations using the proposed repair method. The effect of operators and local search algorithms in ALNS is analyzed considering the solution time and quality.

The remainder of this paper is as follows: The second section includes the problem definition and constraints. The third section presents the general framework of the ALNS algorithm as the method, including the initial solution, operators used, acceptance criteria, and the method itself. The fourth section discusses the study environment, interprets the algorithm's results according to the methods used, and suggests future research directions.

2. Problem Definition.

The vehicle routing problem (VRP) is an optimization problem that aims to determine the lowest-cost delivery routes from a warehouse to a geographically dispersed set of customers. With the rise of electric vehicles, many logistics companies have considered using Electric Vehicles (EVs) in their vehicle fleets to reduce greenhouse gas emissions and lower fees per gram of emissions/km. Due to environmental concerns, EVs have been employed for last-mile deliveries in several places. EVs have advantages such as cheap transportation costs, energy efficiency, and lower emissions of pollutants. EVs are powered by rechargeable batteries. However, EV application faces significant difficulties, such as poor battery energy density, inadequate electric charging and battery switch stations, and longer recharging times. To prevent pointless diversions, route planning should include factors such as limited cruising range, lengthy recharge times, and a lack of recharging facilities. These restrictions add to the complexity and difficulty of the EV route optimization challenge. EVRP studies include various features and constraints. The Capacitated EVRP (CEVRP) is a VRP based on EV's with limited carrying capacity. Objects have a quantity, such as weight or volume, and vehicles have a maximum capacity they can carry. The basic assumptions of the CEVRP are as follows:

1. Each route starts and ends at the depot node.
2. All vehicles in the fleet leave the depot with a full charge
3. The time required for a full charge is known.
4. Each customer node will be serviced by exactly one electric vehicle.
5. Multiple packages can be delivered to a customer.
6. The load-carrying capacity of the vehicle cannot be exceeded.
7. Electric vehicles can visit a charging station between two customers.
8. Each charging station can be visited by more than one electric vehicle.
9. The locations of the charging stations and the travel distance from any node to any charging station are known.

In this study, problem sets with 5, 10, 20, 40, and 60 customers, categorized as Clustered (C), random (R), and randomly clustered (RC), are used on the Osmangazi University campus. The shortest distances between all nodes representing customers and charging stations are obtained using the Dijkstra algorithm. The problem is represented by a graph in Equation 1.

$$G = (V, E) \tag{1}$$

Here, V is the nodes representing the depot, customers, and stations, while E is the edges representing the roads.

3. Hybrid ALNS Algorithm

To tackle challenges such as battery capacity, payload capacity, and charging times in the CEVRP problem and to obtain high-quality solutions within short computation times, the ALNS algorithm is proposed [3].

The ALNS algorithm consists of specific steps and components. First, an initial solution is determined, which is expanded by applying certain destruction and repair operators. A new solution is obtained in each iteration. The new solution is evaluated against the previous solution. S is the current solution, S' is the new solution and S^{best} is the best solution. The steps of the ALNS algorithm are summarized as follows:

1. Creating the initial solution.
2. Generating new solutions using operators.
3. Producing neighbor solutions using Local Search.
4. Evaluating new solutions and accepting them with the Simulated Annealing (SA) acceptance criterion.
5. If the stopping criterion has not been reached, go back to step 2.

In the proposed Hybrid ALNS algorithm, LS operators and SA acceptance criteria have been integrated into the ALNS to improve the solution quality.

3.1. Initial Solution

Proper creation of the initial solution can effectively converge to a good solution. However, it can also be generated completely randomly [11]. In this study, the initial solution considers the payload capacity of the electric vehicle. First, an empty route list is created, and the depot is added. Then, the nearest customer to the depot is added to the route. Continue to add the unserved customer with the earliest delivery date as long as the vehicle capacity is not exceeded. In each addition, the served and unserved lists are updated. If a load issue arises with the last added customer, this customer is removed from the route, and the depot is added at the end of the route list to complete the route.

3.2. Operators and Operator Selection Mechanism

Neighbor solutions are created by destroying and subsequent repair approaches. The variety of operators is significant for obtaining reasonable solutions. Each set of operators has its own weight, and initially all the operators are assumed to be equal weights. In the initial state, the weights are calculated by the formula $\frac{1}{\Omega_+}$ and $\frac{1}{\Omega_-}$, where Ω_+ denotes the set of repair operators, and Ω_- is the set of destroy operators. Also, the local search (LS) operators have equal operator weights. The ratio of the weight of each operator to the sum of the weights of that operator set gives a probability value. The calculation of the likelihood value is given in Equation 2, where *ω(i)* represents the operator weight.

$$P(i) = \frac{w(i)}{\sum_{i' \in \Omega} w(i')} \qquad (2)$$

This process represents the probability of an operator being selected. This structure is called the roulette wheel method. The calculated operator probability is increased in specific iterations according to the performance score. The procedure for finding this score is given in Equation 3.

$$\pi_i = \begin{cases} \pi_1, & if\ S' < S^{best} \\ \pi_2, & if\ S^{best} < S' < S \\ \pi_3, & if\ S' > S \\ 0, & if\ S = \emptyset \end{cases} \qquad (3)$$

The score is set to zero if the operator leads to the best result, improves the existing solution, is an acceptable solution, or fails to find a solution. At the given iteration step, the operator weights are updated with Equation 4.

(4)

When determining the new weight, *w(i)* is the weight before the update, Q_i is the number of times of operator *i* is used, and π_i is the score of operator *i*. The parameter *r* determines the balance that will be dominant between the old weight and the newly calculated score π_i/Q_i. This value is randomly chosen between (0,1) and is essential for parameter optimization. A high value indicates a weight preference for score, and a low value indicates a weight preference for past performance.

Customer Removal Operators: Three customer removal operators are used that are given in Table 1.

Table 1 : Customer removal operators.

Operator	Operator description
Related Customer Removal	Selecting a starting customer removes related customers that close to that customer from the route.
Random Customer Removal	*P* randomly selected customers are removed from the route.
Worst Distance Customer Removal	The customers that will cause the largest cost reduction or are in the highest range are removed from the route.

Customer removal operators select and remove customers according to specific methods. The removed customers are added to the unserved list.

Customer Insertion Operators*:* Two customer insertion operators are used as given in Table 2.

Table 2 : Customer insertion operators.

Operator	Operator description
Greedy Customer Insertion	Calculate the cost of the selected customer when added to each location in the route. It is added to the route with the best cost in a greedy approach.
Regret-2 Customer Insertion	A regret value is calculated for each customer and the customer with the largest regret value is added.

The customer insertion operator selects and adds removed customers to the unserved list.

Route Removal Operators: Route removal operators are given in Table 3.

Table 3: Route removal operators.

Operator	Operator description
Random Route Removal	All routes within the selected random route customers are removed.
Greedy Route Removal	Select the route with the highest cost among all routes and remove all customers on the route.

The route removal operator removes all customers of the selected route and adds them to the unserved list.

Local Search Operators: The Local Search operators are given in Table 4.

Table 4 : Local Search operators.

Operator	Operator description
IntraRelocate	Moving a customer on the same route to another location.
IntraExchange	Relocation of two customers on the same route.
IntraOrOpt	Relocation of several consecutive customers on the same route to another location.
IntraTwoOpt	Relocation of two sub-routes on the same route.

InterRelocate	Relocation of one customer on different routes to another route.
InterExchange	Relocation of two customers on different routes.
InterCrossExchange	Relocation of two sub-paths on different routes.
Inter2Opt*	Reorganizing routes by changing the intersection points of two sub-paths on different routes.

Local search operators aim to improve solution quality with different route operations.

3.3. Acceptance Criteria

Each new solution S' obtained from applying the operators is compared with the current solution *S*. If a better solution is obtained, it is accepted as the current solution. Otherwise, the algorithm accepts the poor solution with a probabilistic approach. The acceptance criterion evaluates the probability value. In this study, the Metropolis Criterion, which is the acceptance criterion of the SA algorithm, is used. It is given by Equation 5 and Equation 6.

$$\Delta E = f(S') - f(S) \tag{5}$$

$$P = e^{(-\Delta E / T)} \tag{6}$$

By calculating the difference between the current and new solution costs, the function $e^{(-\Delta E/T)}$ depending on the temperature value *T* produces a probability value *P*. The *P* value is compared with the generated random value between 0 and 1. If the *P* value is greater than the random value, the worse solution is accepted. In Equation 6, worse solutions are accepted by producing values close to 1 at high-temperature *T*. As the temperature decreases in subsequent iterations, the *P* value converges to 0 and only better solutions are accepted. The pseudo-code is given in Algorithm 1.

Algorithm 1: SA acceptance criteria

```
S  ← current solution
f( ) ← cost function
T  ← temperature
S* ← best solution
acceptance rate ← e^(−(f(S') − f(S))/ T)
until termination criteria then
    S' ← generate new solution
    if f(S') < f(S) or acceptance rate > random(0,1) then
        S ← S'
        if f(S') < f(S*) then
            S* ← S
        end if
    end if
    T ← T * α
```

The temperature is decreased using Equation 7.

$$T(t+1) = \alpha .\ T(t) \tag{7}$$

At each iteration, the temperature is reduced by cooling parameter α.

3.4. Neighborhood Search

In this study, the Hybrid ALNS algorithm is used for the CEVRP. The pseudo-code of the ALNS algorithm is given in Algorithm 2. The performance of ALNS, ALNS_Swap and ALNS_LS are compared for solving CEVRP problem.

ALNS: A methodology in which ALNS includes customer removal, customer insertion and route removal operators.

ALNS_Swap: In addition to ALNS, the method includes *IntraRoute* and *InterRoute* operators that perform swapping in route (intra-route) and between routes (inter-route).

ALNS_ LS: The method uses all eight local search operators in addition to ALNS operators.

Algorithm 2: Hybrid ALNS with LS/Swap operators

```
S  current  ← Get initial solution
S'  new solution, S* best solution ← S
f( ) ← cost function
T  ← temperature
K  ← Predefined route removal iteration interval
Z  ← Weights update interval
for i ← 1 to MaxIterations do
    if i == 0 (mod K) then
        Call route removal operator (RR)
        S' ← Call customer insertion operator (CI)
    else
        Call customer removal operator (CR)
        S' ← Call customer insertion operator (CI)
    end if
    acceptance rate ← e^(−(f(S') − f(S))/ T)
    if f(S') < f(S) or acceptance rate > random(0,1) then
        S ← S'
        new S' ← Call LS operator or Swap operator
        if f(new S') < f(S) then
            S ← new S'
        end if
        if f(S) < f(S*) then
            S* ← S
        end if
    end if
    if i == 0 (mod Z) then
        Update weights based on scores
    end if
    T ← T * α
end for
return S*
```

In the Algorithm 2, current solution (*S)* replaces the initial solution with operators at each iteration. Each iteration uses customer removal and customer insertion operators, respectively. In every *K* iteration, the route is wholly disrupted and reorganized. These operations produce S' solutions. The new solutions are accepted as in Algorithm 1. Swap or LS operators are applied to the accepted solution to obtain new solution S'. This is done to approach the global best. If the new solution S' is better than *S*, the value of *S* is updated. When the best solution is found, *S** is updated.

3.5. Station Insertion

For large size problems, it is observed that the solution time increases significantly when the battery capacity is checked in addition to the load capacity check at each iteration. Therefore, when the specified number of iterations is reached, the station addition operator is used to find the best solution. The need to add a charging station arises before the customer node indicates that the vehicle's battery capacity has been exceeded. The cost calculation of station *i* for each charging station is given by Equation 8 where k_1 is the state of charge from the previous customer to the station, and k_2 is the state of charge from the charging station to the customer point where the battery capacity is exceeded.

$$cost = k_1 + k_2\ ,\ k_1 > 0,\ k_2 > 0 \tag{8}$$

According to the constraint given by Equation 8, k_1 and k_2 should be positive. The approach for station insertion is given in Figure 1.

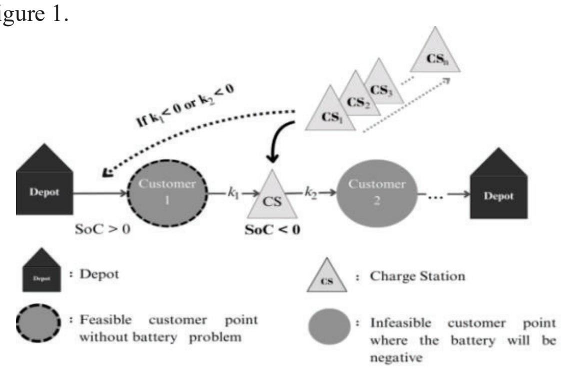

Figure 1: Charge Station insertion approach.

In the example route in Figure 1, the EV has a sufficient state of charge (SoC) for customer 1 but requires additional charging for customer 2. This means an infeasible route. In this case, the route should be made feasible by adding a charging station (CS) between customer 1 and customer 2 to complete the route. If k_1 *<0* or k_2 *<0,* the station insertion is applied to the previous point. The map representation of a C10 problem made feasible by adding a station is given in Figure 2.

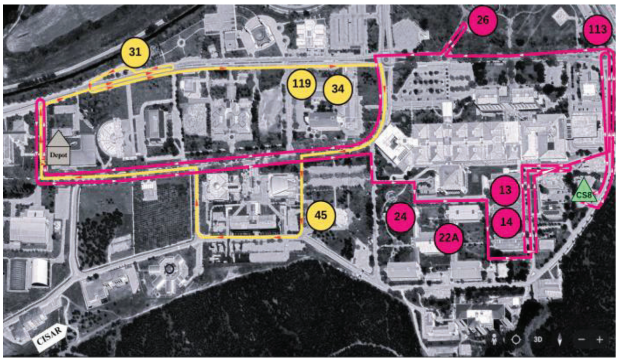

Figure 2 : The representative routes for the problem set C10

As seen in Figure 2, the yellow route was completed without going to the charging station. On the pink route, when passing from customer 113 to customer 26, the charging constraint given by Equation 8 is not met. Therefore, the charging station numbered CS8 between these two customers has been added to the existing route.

4. Comparison of Performances

The solutions are obtained for the given problem sets using ALNS, ALNS_Swap, and ALNS_LS. Their performances are compared in terms of the minimum total distance objective function. The experiments used Python 3.12 with an AMD Ryzen 7 4700U @2.00GHz processor and 8GB RAM. For each problem, the methods are run with 10000 iterations to get results in a reasonable time and a cooling rate of 0.9985, which means the temperature does not drop quickly. Weight update is chosen every eight iterations, and the route removal is achieved every 50 iterations. In our test environment, the battery capacity is set at 3000 kWh to test the EV's charging station visits. In addition, the vehicle capacity is set at 350 kg. The results are given in Table 5.

Table 5 : Comparison of the results.

Problem	Minimum Total Distance (m)			Solution Time (s)		
	ALNS	**ALNS Swap**	**ALNS LS**	**ALNS**	**ALNS Swap**	**ALNS LS**
C05	4506	**4129**	**4129**	4,17	**3,49**	4,08
R05	**5718**	**5718**	**5718**	3,06	**2,19**	2,50
RC05	**4982**	**4982**	**4982**	3,03	4,05	**2,60**
C10	**6497**	**6497**	**6497**	3,79	4,21	**3,32**
R10	**5891**	6278	6055	4,68	**3,68**	3,88
RC10	**6736**	**6736**	**6736**	4,30	3,92	**3,56**
C20	11938	10978	**10443**	6,31	6,12	**5,78**
R20	**11817**	**11817**	**11817**	7,36	**5,69**	6,36
RC20	10065	**10028**	**10028**	**5,87**	6,83	6,75
C40	**17565**	17763	17609	14,49	**12,31**	12,42
R40	**19770**	19996	**19771**	15,31	**11,58**	13,98
RC40	18402	**17867**	**17867**	15,37	**12,72**	14,01
C60	29400	29282	**28709**	27,45	**21,67**	24,50
R60	27617	**27471**	27546	31,64	**22,85**	26,82
RC60	28062	**27817**	28164	30,44	27,43	**25,84**

When the results are analysed in terms of total distance, it is seen that ALNS gives poor results for the datasets with clustered customers, especially in the problem with 20 customers (C20). On the other hand, ALNS_Swap gives worse results on data with clustered customers and the best results on data with random clustered customers. While ALNS_LS gives better results on data with the clustered customers, it is moved away from good solutions on data with random and randomly clustered customers. However, in general, ALNS_LS outperforms the others. For the comparative analysis, the difference with the best value for each problem is given in Table 6.

Table 6 : Differences from the best solutions

Problem	Gap		
	ALNS	**ALNS_Swap**	**ALNS_LS**
C05	377	0	0
R05	0	0	0
RC05	0	0	0
C10	0	0	0
R10	0	387	164
RC10	0	0	0
C20	1495	535	0
R20	0	0	0
RC20	37	0	0
C40	0	198	44
R40	0	226	1
RC40	535	0	0
C60	691	573	0
R60	146	0	75
RC60	245	0	347
Total	**3526**	**1919**	**631**

As seen in Table 6, ALNS had the highest gap value in total and made a difference in the largest number of datasets. ALNS_Swap showed less gap value than ALNS and showed a difference in fewer datasets. Besides, ALNS_Swap has less runtime than the others. ALNS_LS has the lowest total gap value and showed the least difference in the dataset.

According to the results in Table 6, ALNS_LS gives results closer to the known best solutions than the other two algorithms. According to these results, ALNS_LS shows the best performance because it has the lowest total gap value and created the least difference in the dataset. ALNS has the highest total gap value and appears to be the weakest method in terms of performance.

5. Conclusions

This study deals with the CEVRP by using Hybrid ALNS algorithm. The adaptive nature of the ALNS algorithm leaves an open door for experimenting with different methods, allowing us to obtain effective results with operators that can better scan the solution space. The study compares the performance of ALNS, ALNS Swap, and ALNS_LS. ALNS_Swap is derived from ALNS and uses swapping operators. ALNS_LS performs local search with eight additional operators. For the minimum total distance objective function, feasible routes are obtained in the full charging strategy. The results show that using LS and displacement operators gives better results than the average. In particular, it positively affects the result regarding distance and running time.

In future work, it is planned to include charging station addition and removal operators in the operator pool to improve the performance of the algorithm. Station operators will be used within iterations to search the better solutions. Furthermore, different charging strategies and objective functions can be used by considering the customer requirements. The impact of different charging strategies on the solution will be analysed.

Acknowledgement

This paper is supported by the OPEVA project that has received funding within the Key Digital Technologies Joint Undertaking (KDT JU) from the European Union's Horizon Europe Programme and the National Authorities (France, Belgium, Czechia, Italy, Portugal, Turkey, Switzerland), under grant agreement 101097267. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or KDT JU. Neither the European Union nor the granting authority can be held responsible for them. This work is supported by the Scientific and Technical Research Council of Turkey (TUBITAK), Contract No 222N269, project title: "OPtimization of Electric Vehicle Autonomy (OPEVA) "

References

[1] Li, Z. and Xu, H.Y. (2021), "Accounting for 15% of the country's final carbon emissions, how can the transportation industry achieve carbon peak and carbon neutrality?"https://www.yicai.com/news/100976813.html.

[2] A. Juan, C. Mendez, J. Faulin, J. De Armas, and S. Grasman, "Electric Vehicles in Logistics and Transportation: A survey on emerging environmental, strategic, and operational challenges," Energies, vol. 9, no. 2, p. 86, Jan. 2016, doi: 10.3390/en9020086.

[3] S. Ropke and D. Pisinger, "An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows," Transportation Science, vol. 40, no. 4, pp. 455–472, Nov. 2006, doi: 10.1287/trsc.1050.0135.

[4] E. Demir, T. Bektaş, and G. Laporte, "A review of recent research on green road freight transportation," European Journal of Operational Research, vol. 237, no. 3, pp. 775–793, Sep. 2014, doi: 10.1016/j.ejor.2013.12.033.

[5] M. Schneider, A. Stenger, and D. Goeke, "The Electric Vehicle-Routing Problem with Time Windows and Recharging Stations," Transportation Science, vol. 48, no. 4, pp. 500–520, Nov. 2014, doi: 10.1287/trsc.2013.0490.

[6] M. Keskin and B. Çatay, "Partial recharge strategies for the electric vehicle routing problem with time windows," *Transportation Research. Part C, Emerging Technologies*, vol. 65, pp. 111–127, Apr. 2016, doi: 10.1016/j.trc.2016.01.013.

[7] T. Erdelić and T. Carić, "A survey on the electric vehicle routing problem: Variants and solution approaches," Journal of Advanced Transportation, vol. 2019, pp. 1–48, May 2019, doi: 10.1155/2019/5075671.

[8] T. Carić, "Solving the electric vehicle routing problem using a hybrid adaptive large neighborhood search method," Repozitorij Fakulteta Prometnih Znanosti, Oct. 04, 2021. https://urn.nsk.hr/urn:nbn:hr:119:138947

[9] M. Erdem and Ç. Koç, "Analysis of electric vehicles in home health care routing problem," Journal of Cleaner Production, vol. 234, pp. 1471–1483, Oct. 2019, doi: 10.1016/j.jclepro.2019.06.236.

[10] S. T. W. Mara, R. Norcahyo, P. Jodiawan, L. Lusiantoro, and A. P. Rifai, "A survey of adaptive large neighborhood search algorithms and applications," Computers & Operations Research, vol. 146, p. 105903, Oct. 2022, doi: 10.1016/j.cor.2022.105903.

[11] U. Bac and M. Erdem, "Optimization of electric vehicle recharge schedule and routing problem with time windows and partial recharge: A comparative study for an urban logistics fleet," Sustainable Cities and Society, vol. 70, p. 102883, Jul. 2021, doi: 10.1016/j.scs.2021.102883.

[12] Y. Li, H. Chen, and C. Prins, "Adaptive large neighborhood search for the pickup and delivery problem with time windows, profits, and reserved requests," European Journal of Operational Research, vol. 252, no. 1, pp. 27–38, Jul. 2016, doi: 10.1016/j.ejor.2015.12.032. S.