Korean J. Chem. Eng., 40(10), 2396-2406 (2023)
DOI: 10.1007/s11814-023-1491-2

pISSN: 0256-1115
eISSN: 1975-7220

Highly feasible, energy-minimizing and time window-guaranteeing last-mile delivery routes generation based on clustering and local search considering package densities

Hunggi Lee*, Sohyun Park*, Yongbeom Shin, and Dongil Shin*,**,†**

*Smart Engineering Program, Department of Safety and Disaster, Myongji University, Yongin, Gyeonggido 17058, Korea
**Department of Chemical Engineering, Myongji University, Yongin, Gyeonggido 17058, Korea
(*Received 12 November 2022 • Revised 26 April 2023 • Accepted 3 May 2023*)

Abstract−The last-mile, which is the final stage of delivery, has great meaning for both consumers and suppliers. Consumers first form user experiences with a company in the final delivery stage, and the efficient last mile delivery for suppliers must be achieved by guaranteed delivery time window and minimized delivery cost in energy and time. Execution feasibility is also important, because the optimal route is optimal only when it is actually executed by drivers. In this study, we aim to create an optimal route with high feasibility and minimal energy consumption for green delivery. The proposed method minimizes the sum of volume-weighted delivery time of packages and determines the priority of the visit by considering clusters made from the zone ID sequences systematically extracted from the collected delivery routes. The optimal route is generated by determining the order of visits of inter- and intra-clusters through local search based minimization. Case studies using the actual delivery data provided by the Amazon Last-mile Routing Challenge in year 2021 show its efficiency in achieving green delivery.

Keywords: Routing, Clustering, Local Search, Optimization, Energy Consumption Minimization

INTRODUCTION

The size of the global e-commerce market in 2020 was $4.280 trillion, 27.6% up from the previous year, and expected to grow to $6.388 trillion by 2024. Establishing a company's delivery strategy has a great impact on securing competitiveness and securing customer trust. The last-mile is the final stage of delivery and has great meaning for both consumers and suppliers. Consumers are the first to form a user experience with a company in the final delivery stage, and this user experience has a great impact on their loyalty to the company. Efficient last mile delivery for suppliers achieves cost minimization and enables them to acquire loyal customers [1]. Last mile delivery is also the most inefficient part of the current logistics process, and the cost of this process accounts for 53% of the total logistics process. In the past, fast and damage-free delivery was important, so the delivery route was set based on the shortest distance, but now it is necessary to create an optimal delivery route that takes into account all of the reduction in delivery time, cost optimization, customer requirements, and carbon emission minimization. In fact, in the case of UPS, a U.S. delivery company, chose a route that does not turn left in countries where vehicles drive on the right. Through this, various goals such as fuel saving and safety were achieved, as well as shortening the delivery time by reducing the waiting time for signals and creating a route that reduced the possibility of collision with a vehicle coming from the opposite side [2].

Since creating an optimal delivery route has become an important task, most delivery companies create optimized routes by reflecting various factors such as safety and efficiency, but actual drivers tend not to follow the optimized route. Unlike delivery companies, experienced drivers take into account their tacit knowledge of the complex operating environment when serving customers on a daily basis. Leveraging this tacit information to improve route planning is critical to safer, more efficient and sustainable last-mile delivery. Accordingly, Amazon and MIT held the Amazon Last Mile Routing Challenge in 2021, and the challenge can get a better score the more it creates a route that is most similar to the actual delivery route.

Last mile delivery route optimization can be defined as a kind of traveling salesman problem (TSP). TSP is a problem of generating a route that visits all cities once at the minimum cost given the cost of moving distance and required time between several cities and each city. If the global search algorithm is used to solve the TSP problem for N cities, the time complexity is $(n-1)!$. Therefore, as the number of visited points increases exponentially, it is one of the NP-hard. The local search technique is widely used to find the approximate solution of TSP and LKH algorithm that is known to be the most efficient among them, and found many of the best optimal solutions known so far. Dorigo et al. conceived an idea from an ant colony, proposed ant colony optimization, and applied it to various types of TSP [3]. Holland proposed a genetic algorithm inspired by Darwin's theory of evolution [4]. Brady tried to solve TSP through a genetic algorithm for the first time [5], and Nagata et al. solved large-scale TSP by introducing the edge assembly crossover method in the crossover process of the genetic algorithm and suggested how to do it [6]. In this study, our objective was to minimize energy consumption by designing a delivery route that reduces the amount of package and time required. To accomplish this, we conducted our study using the dataset from the Amazon Last Mile Routing Research Challenge and utilized Google's OR-tools with a

†To whom correspondence should be addressed.
E-mail: dongil@mju.ac.kr

local search algorithm during the optimization process.

BACKGROUND THEORIES

1. Traveling Salesman Problem (TSP)

TSP is a problem of finding a route with the minimum cost when there are several cities, and when the cost of moving from one city to another is given in terms of time, distance, etc., visiting all cities once and returning to the starting point. The time complexity of the TSP visiting n points is $(n-1)!$ [7]. Since the optimal solution search time increases exponentially as the number of visited points increases, therefore, recent studies have been conducted to solve TSP using heuristic methods.

2. Local Search Algorithms for TSP Solution

Local search is a heuristic technique for solving computationally difficult problems. It is a method to iteratively find a better solution in the current state using the objective function among several candidate solutions. Representative local search algorithms for solving TSP include simulated annealing (SA), genetic algorithm (GA), and k-opt-based algorithm, which is an edge swapping algorithm [8-10].

2-1. 2-Opt, 3-opt Algorithm-based Optimization

The 2-opt and 3-opt algorithms start with the basic concept of untwisted edges. In the case of 2-opt, as shown in Fig. 1, two random edges are selected, the existing edges are removed from the nodes where the edges are connected, and new edges are connected. At this time, only the case shorter than the original length is selected, otherwise it is skipped. When connecting two new edges, the important thing is to make a complete tour without creating a subtour. In this case, the number of possible 2-opts is only one. Similarly, 3-opt is a method of selecting three edges and selecting a new edge that makes a complete tour [11].

Although the 2-opt method derives an answer faster than the 3-opt method, the 2-opt method is highly likely to fall into the local optimal solution and has a characteristic that the direction of the route is reversed.

2-2. Optimization Based on LKH Algorithm

The multiple use of 2-opt and 3-opt is called k-opt, and the LKH (Lin-Kernighan-Helsgaun) algorithm is a method based on k-opt. It generates an optimal path based on the k-opt, after dividing a large problem into multiple regions. When dividing into multiple regions, methods such as k-means are used. After dividing into several regions, the divided subtours are merged, and k-opt is performed again on the merged tour to make the route corresponding to the local optimal solution close to the global optimal solution [12,13].

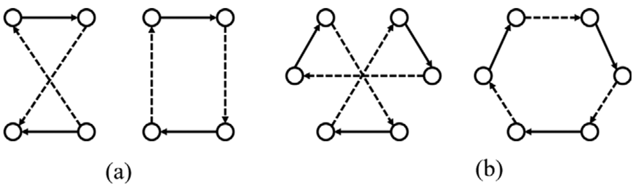


Fig. 1. Untwisted edges after edge swapping using (a) 2-opt and (b) 3-opt.

3. Evaluation of Route Execution Feasibility

The evaluation of the derived delivery route calculates the similarity with the delivery route taken by the actual delivery person through the sequence of deviation (SD) and the edit distance with real penalty (ERP). ERP is a method of mathematically measuring the similarity of two sentences by calculating the minimum number of operations required when converting to another string [14]. The score function provided by the challenge is as follows:

$$\mathrm{Score} = \frac{\mathrm{SD(A, B)} \cdot \mathrm{ERP}_{norm}\mathrm{(A, B)}}{\mathrm{ERP}_c\mathrm{(A, B)}} \tag{1}$$

where A is the actual delivery route, B is the derived optimal delivery route, and SD is the deviation between A and B delivery routes. ERP_{norm} is the travel time penalty for the difference between A and B route, and ERP_c is the number of differences between A and B route. Since the actual route A and the optimal delivery route B each consist of a stop string, the similarity of the string can be measured using ERP. As the difference between the optimal delivery route and the delivery source's actual route increases, the score increases. For randomly generated routes, the average score is 0.8 to 1.2. In this study, we tried to create a route that is likely to be chosen by an actual delivery person through the corresponding score, and this score was defined as feasibility.

4. Energy Consumption Calculation

We derived energy consumption from the carbon emission calculation formula. The basic way to calculate carbon emissions from transport is to multiply actual fuel consumption by CO_2 emission factor. However, if the fuel consumption cannot be accurately estimated, the fuel economy method, the ton·km method, and the rate method are alternatives [15].

In this study, the approximate energy consumption was calculated by multiplying the transport weight by the transport distance as in the ton·km method. However, we used the volumetric weight method to estimate the weight because the Amazon Challenge only provides the dimensions of the products shipped. Also, since the score function provided by the challenge imposes a penalty based on the travel time, the transportation amount was calculated using the travel time instead of the distance.

$$t_{ij} = \text{Travel time}_{ij} \times (\text{Total volume weight} - \text{Package volume weight}_i) \tag{2}$$

where t_{ij} is the energy consumption and Travel time_{ij} is the travel time from delivery point i to j, and Package volume weight_i is the sum of the volume weights of the total packages at delivery point i. In this study energy consumption t_{ij} is calculated by multiplying the value obtained by subtracting the Package volume weight_i from the total volume weight by the travel time.

LAST-MILE ROUTING DATA ANALYSIS

1. Data Definition

In this study, last-mile routing optimization was identified and carried out in the form of adding various constraints to the existing TSP. With the existing TSP, it is difficult to reflect the tacit knowledge of the driver when creating a route with given data. Therefore, it is necessary to add a constraint that can reflect tacit knowledge when creating an actual route. Referring to the data provided by the

Table 1. Provided data

Route data	Coordinates, starting point, zone ID, actual delivery route
Travel times	Travel time between destinations
Package data	Package size, number of packages delivered per branch, time window, delivery vehicle information, delivery date

Amazon Last-mile Routing Challenge in Table 1, we created an optimal route ted using the station, delivery location coordinates, travel time between delivery locations, and the dimensions of the package to be delivered.

2. Selected Data for Route Creation

2-1. Actual Delivery Sequence

The actual sequence is the delivery route actually performed by the driver, and includes the route ID and the stops. The result of plotting the actual delivery route through the python folium library is shown in Fig. 2.

2-2. Station

Station means a local terminal that receives goods from Amazon's fulfillment center and drives them to end customers. As can be seen from the station code the Amazon challenge, 5 cities and 17 stations are given, and a route must be created starting from the station and returning to the station. The data of this challenge represents 6,112 routes from 17 stations in 5 cities: Austin, Boston, Chicago, LA, and Seattle. Station names are 'DAU1', 'DBO1', 'DBO2', 'DBO3', 'DCH1', 'DCH2', 'DCH3', 'DCH4', 'DLA3', 'DLA4', 'DLA5', 'DLA7', 'DLA8', 'DLA9', 'DSE2', 'DSE4', 'DSE5'.

2-3. Package Data

Package data contains information about the goods to be delivered, including dimensions, date of the product to be delivered, the service time it took to deliver, and data on whether the delivery was successful or not. In this study, to minimize the amount of transport, the size of the packages was converted into a volumetric weight commonly used in transport. The time window of the package data represents the customer's preferred delivery time zone and represents a specific time range. The time window was judged to have a significant influence on the route creation, but since the time window was so wide that there were cases where the time window was 12 hours, there was no significant difference compared to when the route was created without consideration. However, in this study, a path was created so that it could be reflected even if the range of the time window is narrow.

2-4. Route Data

Route data contains information about the delivery point: the coordinates of the delivery point, the starting point, and the zone ID, the departure date, and the capacity of the vehicle. In particular, in the case of zone ID, it is assigned to each delivery point, and as a result of comparing the actual sequence with the route data in the order of the actual delivery sequence, most consecutive points consisted of the same zone ID. Through this, it was discovered that a kind of cluster was formed based on the zone ID.

2-5. Travel Times

Travel times data includes time values from one delivery point to another. In this study, the value required to minimize the target transportation volume is distance, but we used travel time data to

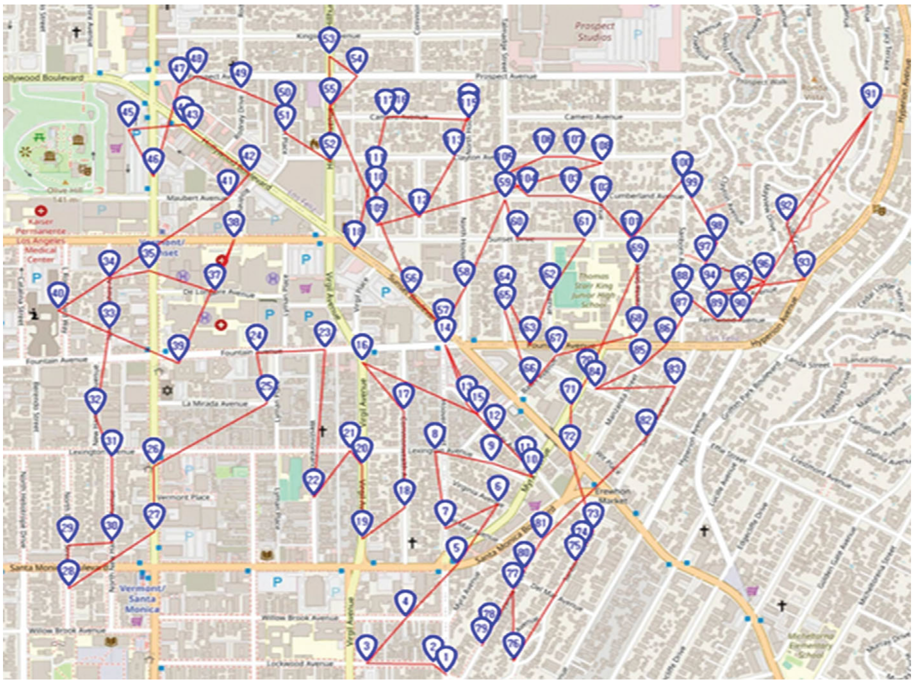


Fig. 2. Actual delivery route.

indicate the distance because a travel time penalty was given to the score function. Therefore, assuming that the speed of the delivery vehicle is constant, the amount of transportation was calculated using time, not distance.

PROPOSED METHOD

The proposed method is a kind of cluster-first, route-second (CFRS) method. CFRS is a method of clustering all delivery destinations and finding an optimized route for each cluster [16]. The zone mentioned later refers to the cluster of CFRS. Zones are set by delivery companies according to specific rules, such as distance standards, capacity that can be delivered at one time, and apartment complexes. This study does not deal with the zone creation method, but assumes that it has already been set according to a specific rule, and that the actual delivery data that can extract the delivery order between zones is accumulated. Using this, we propose a method to solve the feasible last mile delivery TSP problem.

1. Objective Function and Overall Flow Chart

For last-mile delivery, the TSP formula is modified as below:

$$\min \sum_{i=0}^{n} \sum_{j\neq i, j=0}^{n} t_{ij} x_{ij} \tag{3}$$

Subject to

$$t_{ij} = \text{Travel time}_{ij} \times \text{Package volume weight}_i \tag{4}$$

$$\sum_{i=0}^{n} x_{ij} = 1 \quad \forall i \in V \tag{5}$$

$$\sum_{j=0}^{n} x_{ij} = 1 \quad \forall j \in V \tag{6}$$

$$\sum_{i}\sum_{j} x_{ij} \leq |S| - 1 \quad \forall S \subset V,\ 2 \leq |S| \leq n-2 \tag{7}$$

$$t_{ij} \begin{cases} t_{ij} + M & \text{if i and j belong to different zone} \\ t_{ij} & \text{otherwise} \end{cases} \tag{8}$$

$$l_i \leq Tw_i \leq u_i \tag{9}$$

where t_{ij} is the weighted delivery time as shown in Eq. (4), i, j are delivery points, x_{ij} is 1 when driving from city i to city j, otherwise 0. V means the set of all delivery points {0, 1, … n}, and the objective function of the basic TSP that minimizes the delivery time is (3). Constraints (5) and (6) mean that the next visit point and the point before arrival must be one, and constraint (7) removes the subtour that does not visit all stops. In constraint (8), set the M value to a sufficiently large number to set priority when visiting different zones. Constraint (9) is a time window constraint that limits the minimum visit time l_i and maximum visit time u_i when visiting city i [17].

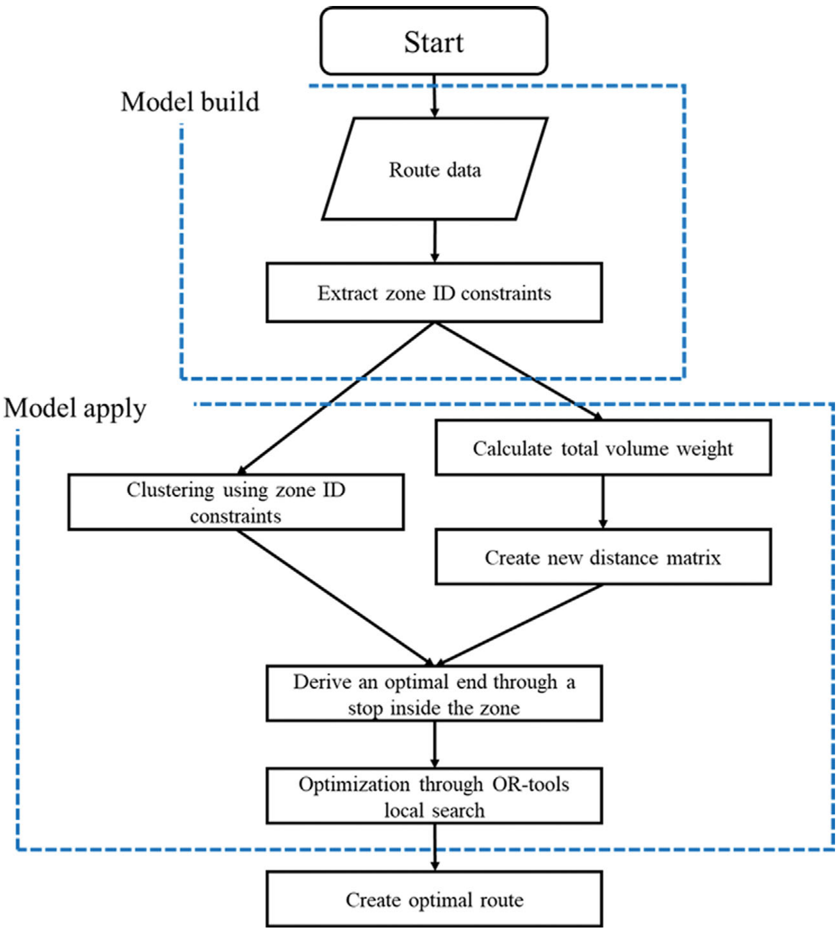


Fig. 3. Flowchart of route creation.

The overall flow chart of the proposed model is shown in Fig. 3. Find the zone ID constraint in the model build process and apply it to 13 routes in the model apply process. In the model apply process, clustering is performed using the zone ID sequence derived from the build process, the priority is determined, and a new distance matrix is constructed by multiplying the travel time by the volume weight calculated using the package volume. Optimized routes were generated using a Google OR-Tools routing model based on zone ID sequences and distance matrices. Google OR-tools is an open-source software that supports optimization programming. In particular, in the routing model, it is possible to solve TSP that reflects various constraints, and optimization is possible by inputting each visit point and distance or travel time as a matrix.

2. Execution Feasibility Improvement by Adding Priority Visit Constraints after Clustering Based on Zone ID

2-1. Extraction of Zone ID Sequences from the Collected Delivery Routes

LKH-AMZ, the best solution method so far, creates a route that minimizes the travel time, and its execution feasibility is obtained by considering the prespecified zone ID sequence.

Similarly, in this study, zone ID-based clustering was used. As a result of comparing the route data in the order in which the actual sequence was delivered, it was found that the delivery point forms a kind of cluster based on the zone ID because most consecutive points have the same zone ID. Each zone seems to be set based on the distance, and the study was conducted assuming that the zone ID acts like a postal code indicating a kind of region. Zone IDs were not assigned to all points and missing values were present. In the case of missing values, the zone ID of the nearest point was assigned based on the distance.

In the zone ID example of Fig. 4, there are English capital letters and numbers at the positions of A, 0, and B, and there is a specific clustering pattern to visit. The capital letter at position A is identified as the largest cluster, the position at 0 as a smaller cluster, and the capital letter at position B as another cluster belonging to A-0. A-2.xA, A-1.xA. It can be seen that the delivery point is visited in the order of A-1.xB and A-1.xC. It can be seen that A-2

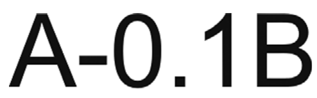


Fig. 4. Zone ID example.

and A-1 are tied together, and in A-1.xA, it can be seen that the number in position 1 in Fig. 4 changes while the capital letter at position B in Fig. 4 is fixed. The number of positions of 1 is a bit irregular, but when moving to the next cluster, there is a specific pattern. If the previous cluster proceeds in the order of 1, 2, 3, the next cluster proceeds in the order of 3, 2, and 1. That is, the number at the position of 1 can be said to be the smallest unit cluster including the delivery point, and the entire zone ID can be said to be a symbol representing clusters of different sizes.

However, we wanted to learn a repeating sequence from the data of the build process. Because following this pattern too much can lower your score. If there is a previous sequence for a new input value, the most repeated sequence is selected, and if there is no previous sequence, the Zone ID sequence is extracted by applying the above pattern. Clustering is carried out step by step according to the order of zone IDs mentioned above, and then the order of visits inside the cluster is determined using OR-tools as shown in Fig. 5. In general, applying clustering to TSP has the advantage of reducing computation time. This study uses zone ID constraint-based clustering to help prioritize visits. Even non-Amazon datasets often require setting up a place to visit first, such as the same apartment complex, to solve the last mile problem with many delivery points.

2-2. Zone ID Sequences as Priority Visit Constraints

Therefore, in this study, in the 'model build' process, repeated Zone ID sequences were recorded according to 17 stations appearing on 6,100 routes excluding test data. In the 'model apply' process, given test data, it contains the zone ID of the new data and acquires the most repeated sequence first. If a previous sequence exists for a new input value, the most repeated sequence is selected, and if there is no previous sequence, the zone ID sequence is extracted by applying the rule mentioned in 4.2.1. The result is shown

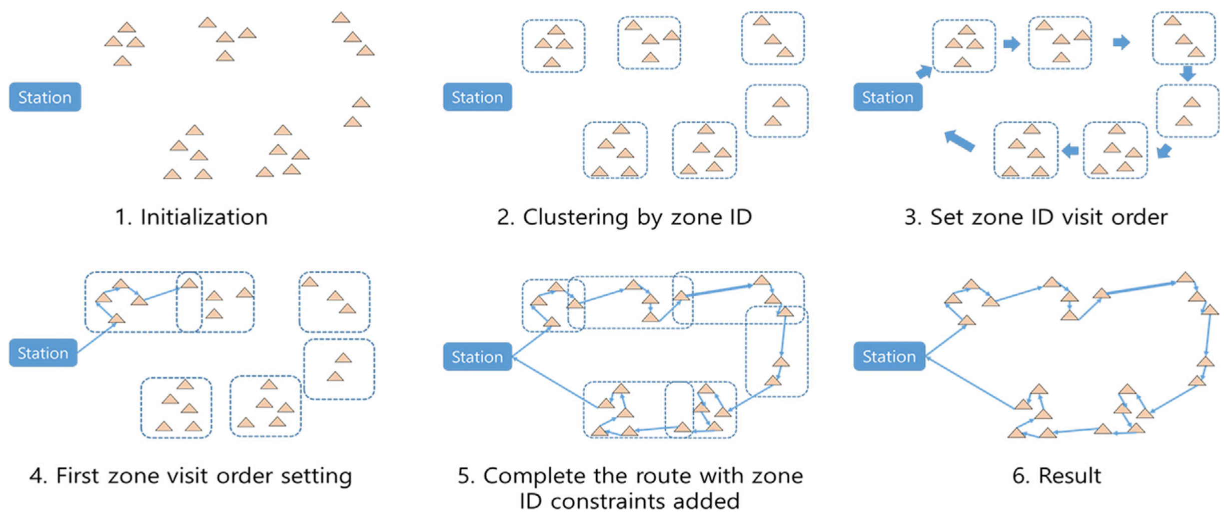


Fig. 5. Algorithm sequence.

	index	zone_id	zone_rank
0	AE	B-29.1B	4
1	AG	B-29.1C	5
2	AI	B-29.3H	21
3	AL	B-29.2E	12
4	AO	B-29.3B	2
..	...	...	...
169	YU	B-29.1J	22
170	YW	B-29.3A	1
171	ZK	B-29.2C	6
172	ZS	B-29.2E	12
173	ZX	B-29.3A	1

Fig. 6. Example of zone ID sequence derivation result.

in Fig. 6.

3. Minimization of Delivery Time Multiplied by the Volume Weight of the Delivered Product for Minimal Energy Consumption

3-1. Volume Weight

First, we created a constraint on the order of cluster visits using the zone ID, and then generated a new distance matrix by multiplying the travel time between each point by the volume weight of the package. The weight of the delivered item is needed to calculate the volume, but for this challenge, we only provided the dimensions of the delivered item. Therefore, the volume weight was used instead of the weight of the delivered product using the volumetric weight method used in transportation. The commonly used formula for volume weight is as follows:

$$V=(\text{length}\times\text{width}\times\text{height})\div 5000 \tag{10}$$

We weighted the distance matrix by subtracting the volume weight of products delivered at each delivery point from the total volume weight. In the case of 'RouteID_fffd257c-3041-4736-bbe7a-5efea8af1173' in the model apply process, total volume weight of the delivered goods for the entire route is 420.3396 kg. The value of 4.7254 was multiplied by all the travel time values corresponding to 'JN' of the distance matrix to give weight. An example of a new distance matrix generated by the above method is shown in Fig. 7.

3-2. Minimize Energy Consumption

A local search was performed based on the Google OR-tools to generate routes that minimize defined weighted delivery times using the generated distance matrix. Computed by adding the last point of the previous cluster and the starting point of the next cluster to the matrix. Set the route so that the last point of the previous cluster is at the beginning of the array and the starting point of the next cluster is the last point. A final route was generated by arranging the generated route in reverse order. Optimization was performed with the OR-tools routing model using the generated distance matrix and clustered forwarding points. The local search metaheuristic

	AE	AG	AI	AL	AO	...	YU	YW	ZK	ZS	ZX
AE	0.0	160.8	650.4	435.1	165.3	...	669.2	149.6	221.2	458.3	125.7
AG	181.2	0.0	615.4	400.1	288.3	...	634.2	272.1	78.4	423.3	248.2
AI	624.0	604.7	0.0	273.5	620.6	...	235.0	513.0	577.7	264.6	536.9
AL	437.7	402.3	278.2	0.0	526.3	...	347.6	504.6	369.5	67.3	486.7
AO	168.1	265.8	601.6	502.2	0.0	...	705.5	81.7	326.7	525.9	57.8
..	...	...	...	...	...	...	...	...	...	...	...
YU	679.2	643.8	238.3	375.0	718.9	...	0.0	617.1	611.0	411.7	641.0
YW	152.4	250.1	513.7	506.4	101.7	...	623.7	0.0	310.5	529.6	23.8
ZK	234.4	71.2	587.0	371.7	320.7	...	605.8	305.0	0.0	394.9	281.1
ZS	480.3	444.9	256.7	66.7	568.9	...	390.2	504.3	412.1	0.0	528.2
ZX	128.5	226.2	537.6	482.5	77.8	...	647.6	23.8	286.6	505.7	0.0

	AE	AG	AI	AL	...	YW	ZK	ZS	ZX
AE	0	760884	2620264	1837964	...	639949	984278	2016847	539589
AG	672548	0	2529167	1682626	...	1046047	297795	1860801	946085
AI	2723754	2577180	0	1165049	...	2151280	2458246	1075011	2251368
AL	1784636	1641078	1121806	0	...	2077085	1524590	273581	1979055
AO	671637	1171403	2521584	2138430	...	413221	1303048	2311520	316112
..	...	...	...	...	...	...	...	...	...
YU	2811078	2664056	987154	1460148	...	2619949	2544757	1639095	2720344
YW	628539	1143219	2155354	2120061	...	0	1281448	2118801	99994
ZK	929210	329340	2426785	1552185	...	1304339	0	1731138	1203941
ZS	1921154	1774437	1109180	282115	...	2220037	1655386	0	2119850
ZX	528292	1043135	2256484	2045503	...	100026	1181407	2219920	0

Fig. 7. Distance matrix corrected by multiplying volume weight by travel time.

```
{'num_vehicles': 1,
 'distance_matrix': [[0, 0, 580946, 676231, 344621, 332448, 223311, 182595, 54148, 268645, 394573, 2147483647],
                     [0, 0, 580946, 676231, 344621, 332448, 223311, 182595, 54148, 268645, 394573, 2147483647],
                     [550315, 550315, 0, 101197, 235849, 560351, 337047, 524807, 495953, 610532, 617223, 2147483647],
                     [641323, 641323, 75327, 0, 328361, 463201, 429075, 615937, 587221, 543106, 543939, 2147483647],
                     [315540, 315540, 236130, 313019, 0, 439908, 101678, 289911, 260919, 376044, 468479, 2147483647],
                     [378328, 378328, 550108, 474773, 430069, 0, 310444, 160603, 324518, 79059, 198270, 2147483647],
                     [213549, 213549, 357035, 433812, 120829, 337735, 0, 187957, 159008, 273964, 399828, 2147483647],
                     [220327, 220327, 509061, 585861, 272786, 149402, 151501, 0, 165770, 85612, 211514, 2147483647],
                     [54201, 54201, 526892, 603783, 290337, 278152, 168908, 128151, 0, 214286, 340337, 2147483647],
                     [301991, 301991, 589761, 543333, 354275, 63158, 233394, 81981, 247616, 0, 120043, 2147483647],
                     [410179, 410179, 534943, 442309, 462338, 250363, 341746, 190693, 355933, 186938, 0, 2147483647],
                     [2147483647, 2147483647, 2147483647, 2147483647, 2147483647, 2147483647, 2147483647, 2147483647, 2147483647, 2147483647, 2147483647, 0]],
 'depot': 11,
 'sequence': ['BG', 'BG', 'LD', 'MG', 'NB', 'NI', 'RJ', 'RN', 'WI', 'WT', 'XJ']}
```


Fig. 8. Clustering result using zone ID sequence to be applied to OR-tools.

was used as the search parameter, and the time constraint was set to 1 second because fewer than 10 points were optimized. Fig. 8 shows an example of clustered points and the modified distance matrix between them to apply to the OR tools.

CASE STUDY

1. Amazon Last-mile Routing Challenge

Delivery companies such as Amazon create routes that optimize safety, experience, sustainability, and efficiency, but actual drivers tend to deviate from the calculated route and use factors that are not reflected in the optimization model to last-mile [18]. Therefore, Amazon and MIT held the Amazon Last-mile Routing Challenge from March 15 to June 18, 2021, and the challenge was to find the most efficient route as a type of TSP. The provided data is divided into two major categories: the model build process and the model apply process. In this study, a certain zone ID pattern was found in 6,000 routes provided in the build process and applied to 13 routes in the model apply process. Most of the teams that participated in the challenge focused on predicting the zone ID sequence, and did not use all of the provided data; therefore, we tried to conduct research in a way that minimizes energy consumption by using the volume of package data. Using the travel time and package size provided by the challenge to achieve the goals of zone ID sequence prediction and energy minimization, we outperformed other teams that only reflected zone ID predictions.

Table 2. Mean score

Clustering	Clustering-Volume weight	Existing best (LKH-AMZ)
0.01434	0.00796	0.01198

2. Results

As seen in Table 2, when a route was created using only clustering using the zone ID constraint, the average score was about 0.0143, and it was confirmed that the route created using only the constraint was very similar to the actual delivery route. For 'RouteID_fffd257c-3041-4736-bbe7a-5efea8af1173', which had the highest score right after clustering, the route created right after clustering was 0.0025. In the case of clustering and weighting in the same path, it was confirmed that the score improved to about 0.0015. In the same path, the score of LKH-AMZ, which is the existing best solution, is 0.0020, showing a value in between.

Through this, it was confirmed that even if only the zone ID sequence was accurately predicted, an excellent path required by the challenge could be generated. In addition, when LKH-AMZ was used, it was confirmed that a path close to the shortest path was generated even if the zone ID could not be accurately predicted. In

Table 3. Route generation result (energy minimization route generated in this study vs. LKH-AMZ generation route)

Route ID	Delivery points	Total volume weight	Score (feasibility)		Delivery time		Transport volume (time (s) ×volume weight (kg))	
			Proposed	LKH-AMZ	Proposed	LKH-AMZ	Proposed	LKH-AMZ
15baae2d-bf07-4967-956a-173d4036613f	193	758.40	0.005931	0.005138	11,833.5	11,765.8	90,300,930	93,638,563
3f166f0e-fd2e-47ab-96a0-6cbc99cc6eef	166	477.99	0.003076	0.002664	11,453.0	11,371.5	53,466,290	58,078,315
5486294a-503f-4346-b8a9-862e988cbe7c	151	455.35	0.002214	0.011459	13,685.6	13,402.2	63,880,382	63,704,610
693060a6-88bb-4324-9e9c-925d5240263c	168	476.95	**0.028327**	**0.019886**	10,792.7	10,344.5	50,223,827	53,733,531
7f5d87f0-c39f-434f-bf3f-b159ef321909	182	679.58	0.002265	0.025431	13,303.4	13,376.9	88,296,381	105,394,929
9475872b-287f-4c2c-8e29-887766a4e090	187	585.39	0.002218	0.030303	12,116.1	12,031.5	70,128,988	77,530,243
a8f0009d-e50a-49c9-84d3-f9885ad14a54	140	363.56	0.007433	0.044267	10,706.7	10,457.5	39,542,905	40,738,747
bcc07fea-86d2-41e4-9a58-cfc78956dcc7	140	300.74	0.006367	0.006849	10,445.6	10,362.1	31,666,817	36,473,963
d1a8c3dd-fa67-455c-a68d-af2fd6aa5d91	154	655.97	0.028571	0.076737	10,031.3	9,655.3	68,370,661	80,005,971
e6687a05-2453-4edc-b86c-7558ab6d93f6	105	517.52	0.004982	0.004728	5,980.4	5,839.7	31,731,116	39,095,158
2b8df66d-fcd4-438e-931c-3b84b36a5c6b	133	675.44	0.004318	0.089490	10,904.3	10,403.4	74,413,237	77,160,912
f3261fad-5f97-44f6-ae7f-cf169f5d6452	158	540.12	0.005741	0.017077	10,698.2	11,023.3	59,304,606	68,983,656
fffd257c-3041-4736-be7a-5efea8af1173	174	420.33	0.002069	0.001550	12,309.2	12,212.5	50,900,484	52,668,450

this case, it was confirmed that better results could be generated because the travel time penalty for the ERPnorm, that is, the path difference between A and B, was minimized in the score function provided in the challenge. Therefore, in this study, the result generated through the simple greedy algorithm after clustering was changed to the local search basis to further reduce the travel time. In addition, when checking the actual delivery route, it was confirmed that there are often cases where people visit a different place before a point that is close to each other. Therefore, it was judged that there would be additional constraints, and a new method to minimize energy consumption was suggested, and an excellent route could be created through the proposed method.

The result of creating 13 routes in the model apply process is shown in Table 3. The route proposed in 'RouteID_fffd257c-3041-4736-bbe7a-5efea8af1173' is the same as in Fig. 10, and it was confirmed that it proceeds similarly to the actual route and LKH algorithm route in Fig. 9 and Fig. 11. In the case of transport volume, when the volumetric weight was added, it was 50,900,484, and in the case of LKH AMZ, it was 52,668,450. In the case of movement time, when weight was added, it was 12,309.2 seconds, and in the case of LKH AMZ, it was 12,212.5 seconds, confirming that there was no significant difference.

When examining 'RouteID_5486294a-503f-4346-b8a9-862e988cbe7c', which is one of the routes with the highest score difference from LKH-AMZ, it was confirmed that there is some difference in the first visit clustering method. The actual route of this particular

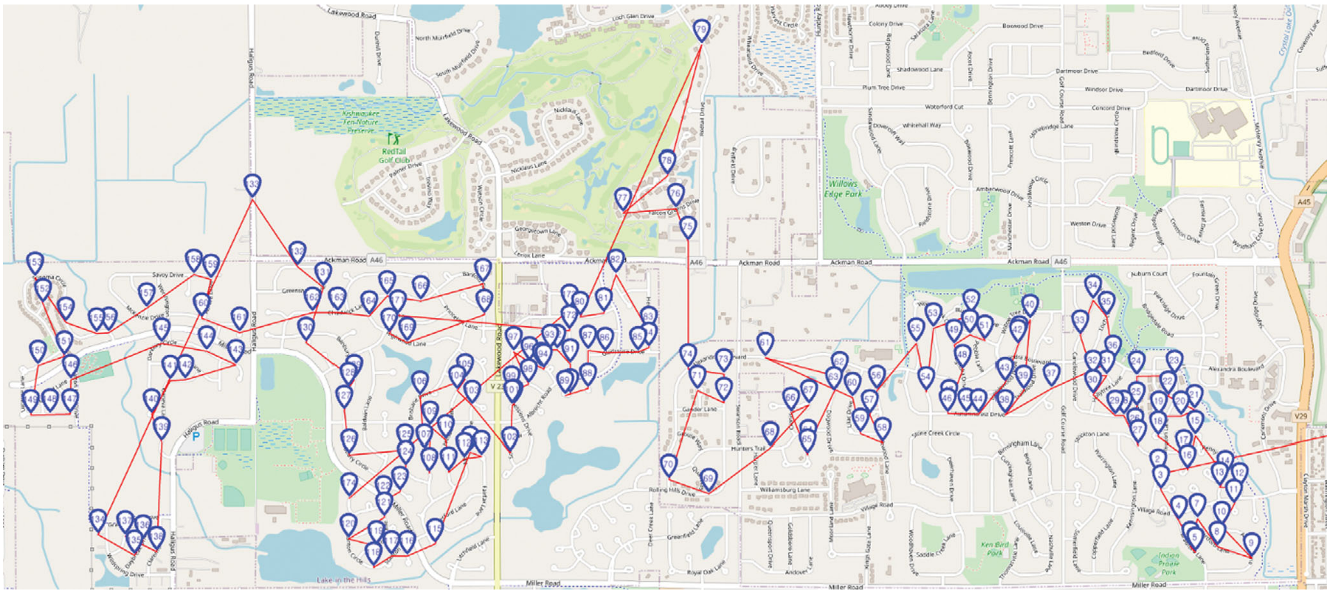


Fig. 9. Actual route 'RouteID_fffd257c-3041-4736-bbe7a-5efea8af1173'.

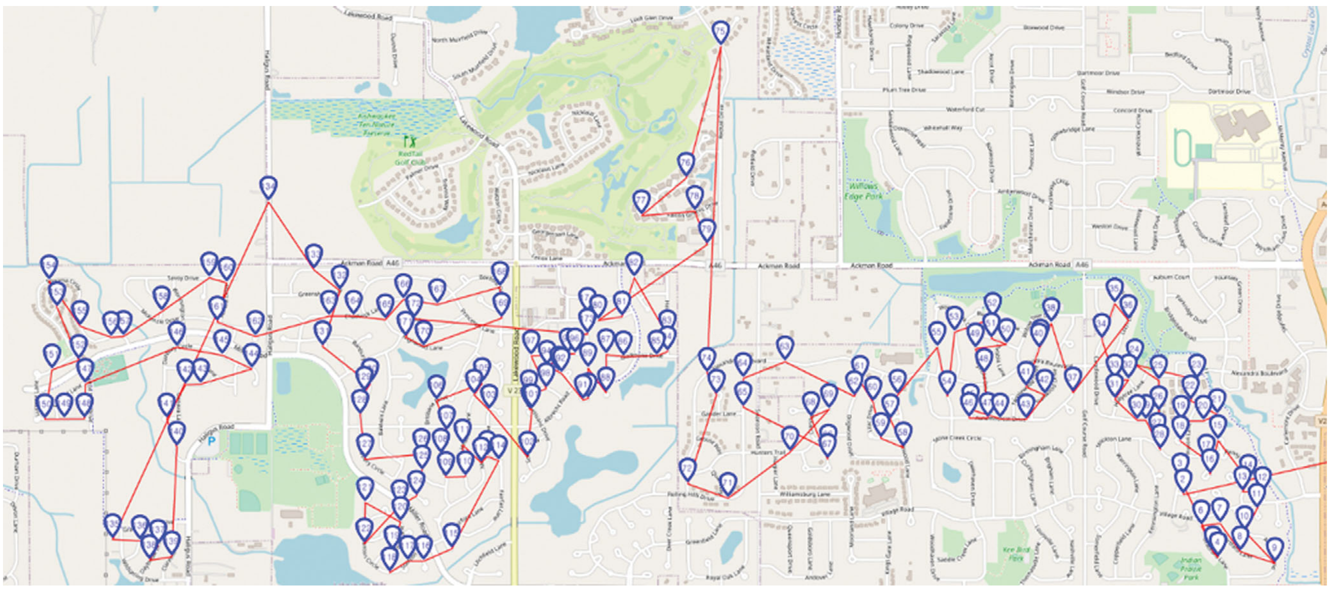


Fig. 10. Proposed route 'RouteID_fffd257c-3041-4736-bbe7a-5efea8af1173'.

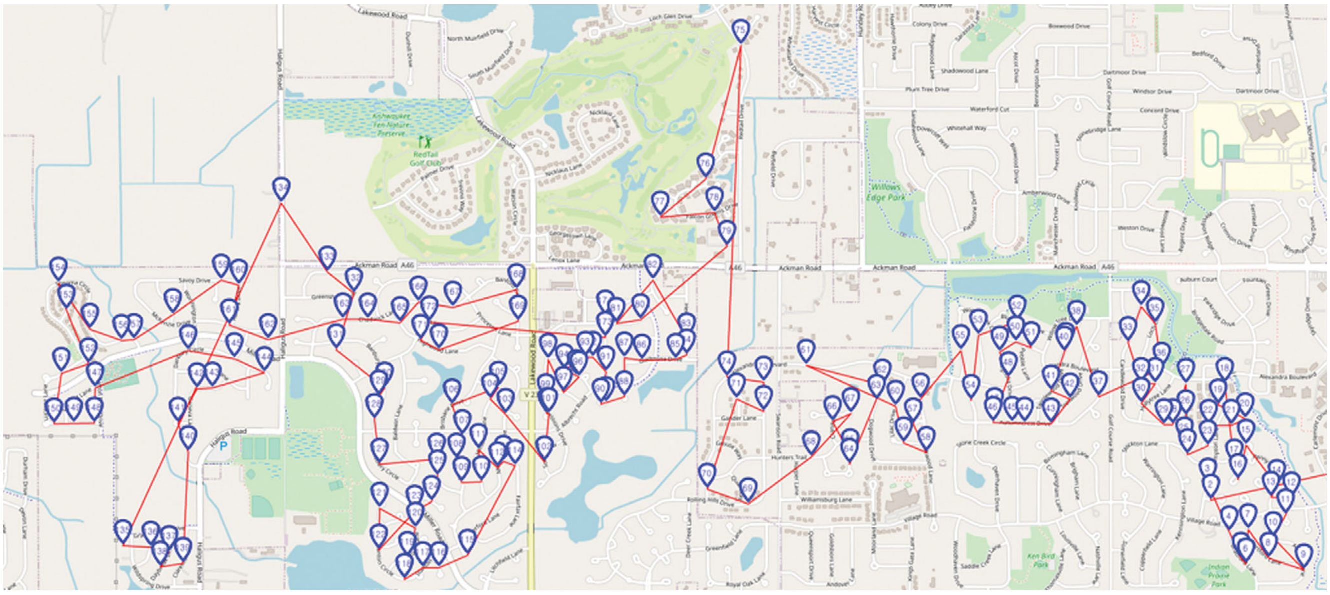


Fig. 11. LKH-AMZ route 'RouteID_fffd257c-3041-4736-bbe7a-5efea8af1173'.

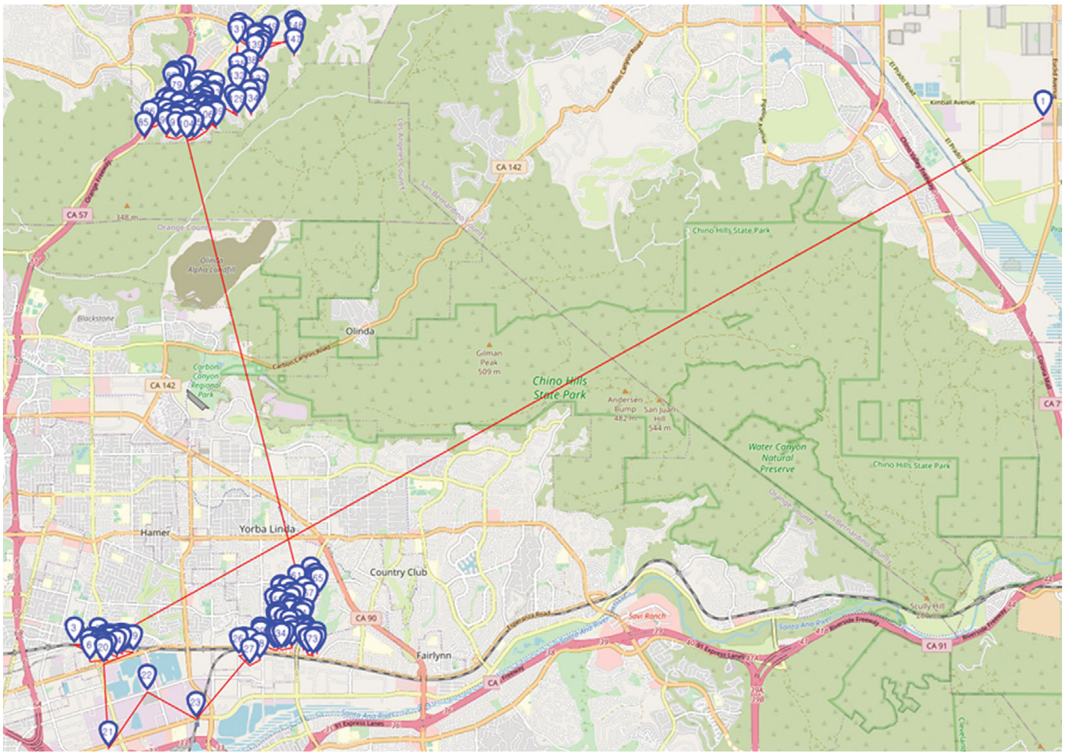
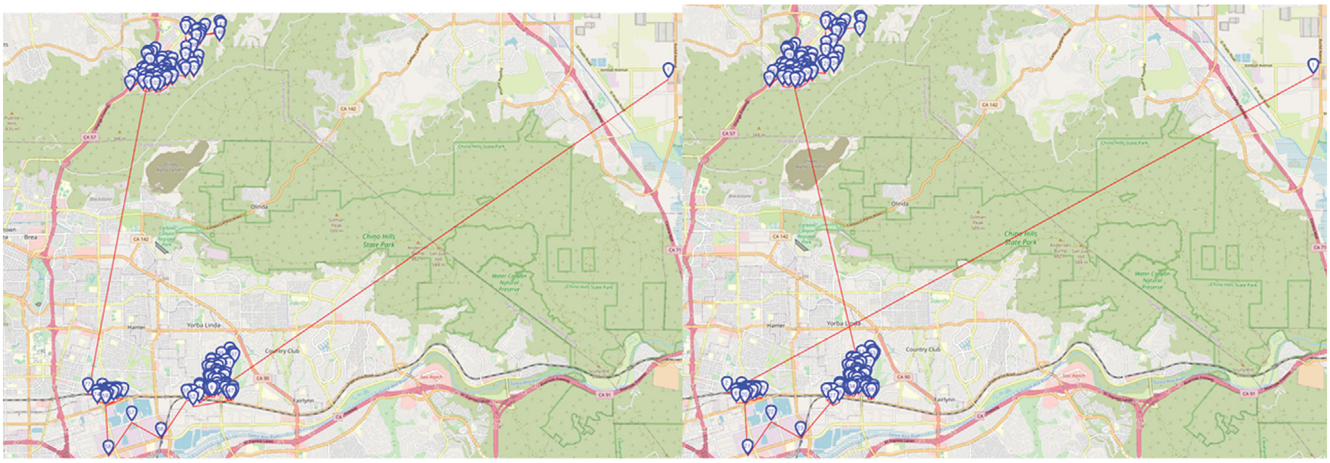


Fig. 12. Actual route 'RouteID_5486294a-503f-4346-b8a9-862e988cbe7c'.

path is the same as shown in Fig. 12, while the routes obtained by applying each algorithm are depicted in Fig. 13. In the case of LKH-AMZ of Fig. 13(b), it seems that there is a tendency to focus a little more on the shortest distance than the algorithm proposed in this study, and it was confirmed that it is focused on reducing the travel times penalty rather than accurately predicting the zone ID. Accordingly, we propose a path that is about 1% faster than the path proposed in this study by about 155 seconds on average in all paths. However, the average score was 0.011989, confirming that the proposed method of this study, which score was 0.00796, produced a better route on average.

Through the derived route, the proposed method of this study was able to reduce the amount of transport, and it was confirmed that the travel time was not significantly different from the current best solution. In addition, by using the zone ID sequence derived from the existing route and the weighted travel time value, we suc-



(a) (b)

Fig. 13. Routes generated by (a) LKH-AMZ vs. (b) proposed method (RouteID_5486294a-503f-4346-b8a9-862e988cbe7c).

ceeded in creating a route more similar to the actual route.

DISCUSSION

Among TSP-related studies, no dataset larger than this one based on actual delivery cases exists. This study is unique in generating feasible routes that actual delivery agents are likely to choose and, therefore, can demonstrate more meaningful research results in real-world situations. Thus, a feasible route means that our proposed model and the route chosen by the actual drivers are very similar in the actual delivery situation. In this study, the zone ID sequence was predicted using the historical data of the model build process, and the weighted travel time was used to optimize and successfully create a path similar to the actual path. Unlike previous studies, we obtained energy consumption simply through volumetric weight. Based on this approach, we designed an algorithm that was able to achieve a result superior to the best known value. Referring to the 'Just pass through' case, the first team in the challenge with the original best solution, the original LKH had a score of 0.07, and the score when using the LKH-AMZ model with domain ID constraints was 0.011989. Since the LKH-AMZ model added the zone sequence-based clustering condition to the existing LKH model, it was confirmed that the feasibility increased according to how accurately the zone sequence was predicted. From this, it can be inferred that Amazon planned the delivery route by dividing the zones. In addition, in the case of the time window, which was considered as one of the important constraints, the given time interval was as long as 12-24 hours in many cases. In the case of moving along the zone sequence, it was confirmed that most of them were satisfied, so it is not a constraint that significantly affects the score in this challenge. In this study, referring to the case of UPS, a US delivery company that tried to achieve the goal of saving fuel, it was attempting to create an eco-friendly route by minimizing energy consumption. In addition, the zone ID sequence prediction of the Amazon problem was also performed, creating a highly feasible, economical and eco-friendly route. As a result, it was very similar to the actual delivery route and was able to generate better results than the existing best solution. Thus, it is judged that Amazon's last-mile delivery also tends to reduce energy consumption by scheduling by reflecting the size or weight of the delivered product.

CONCLUSIONS

With ever-increasing size of the global e-commerce market, the establishment of a company's delivery strategy and the execution have a great impact on securing competitiveness and customer trust. In solving the challenging inefficient last-mile problem in the current delivery process, this study proposes a new method that generates routes guaranteeing minimum energy consumption and high execution feasibility and shows its top-notch efficiency against the actual delivery data provided by the Amazon Last-mile Routing Challenge in year 2021.

LKH-AMZ, the best solution method so far, creates a route that minimizes the travel time, and its execution feasibility is obtained by following the prespecified zone ID sequence. However, the proposed method minimizes the sum of volume-weighted delivery time of packages and determines the priority of the visit by considering clusters made from the zone ID sequences systematically extracted from the collected delivery routes. The optimal route is generated by determining the order of visits of inter- and intra-clusters through local search based minimization of the sum of volume-weighted delivery time of packages.

Compared to LKH-AMZ, the proposed method shows reduced energy consumption by 9%, improved delivery feasibility by 40%, with only 1% increase of the total travel time on average for the given 13 routes of the challenge validation set. Although the proposed method could generate routes with higher feasibility, there was one certain case where LKH-AMZ scored better, because both methods failed to generate good feasibility routes and the score of LKH-AMZ resulted better when multiplied by the travel time penalty.

ACKNOWLEDGEMENTS

This work was supported by Korea Environment Industry &

Technology Institute (KEITI) through Advanced Technology Development Project for Predicting and Preventing Chemical Accidents, funded by the Ministry of Environment (MOE) (2022003620005) and supported by Korea Institute for Advancement of Technology (KIAT) through Smart Digital Engineering Education and Training for Lead Engineer project (P0008475-G02P04570001901) funded by the Ministry of Trade, Industry and Energy (MOTIE). We would also like to thank Ryan Brown, Vice President of Coupang, for providing ideas for this study.

REFERENCES

1. Samsung Securities Co., Ltd., The Future of Retail Last Mile Delivery Service, *https://www.samsungpop.com/mobile/invest/poptv.do?cmd=fileDown&FileNm=uma_200214.html* (2020).
2. Jacopo Prisco, Why UPS trucks (almost) never turn left, *CNN, https://edition.cnn.com/2017/02/16/world/ups-trucks-no-left-turns/index.html* (2017).
3. M. Dorigo, M. Birattari and T. Stutzle, *IEEE Comput. Intelligence Mag.*, **1**(4), 28 (2006).
4. J. H. Holland, *Sci. Am.*, **267**(1), 66 (1992).
5. R. M. Brady, *Nature*, **317**, 804 (1985).
6. Y. Nagata and S. Kobayashi, *INFORMS J. Computing*, **25**(2) (2012).
7. B.-r. Moon, *Easy-to-learn genetic algorithms evolutionary approach*, Hanbit Academy (2008).
8. J. D. C. Little, *Oper. Res.*, **11**(6), 972 (1963).
9. S.-h. Zhan, *Comput. Intelligence Neurosci.*, **2016**(5), 1 (2016).
10. S. Mirjalili, Evolutionary Algorithms and Neural Networks: Theory and Applications, Springer (2018).
11. G. A. Croes, *Oper. Res.*, **6**, 791 (1958).
12. S. Lin and B. W. Kernighan, *Oper. Res.*, **21**(2), 498 (1973).
13. Keld Helsgaun, Efficient Recombination in the Lin-Kernighan-Helsgaun Traveling Salesman Heuristic, *Parallel Problem Solving from Nature-PPSN XV : 15th International Conference*, Coimbra, Portugal, Part I (2018).
14. D. Zhang, *EURASIP J. Adv. Signal Process.*, **2010**, 303140 (2010).
15. J.-H. Kim, *J. Korean Soc. Transport.*, **29**(2), 25 (2011).
16. P. M. Thompson and H. N. Psaraftis, *Oper. Res.*, **41**(5), 935 (1993).
17. G. Laporte, *Eur. J. Oper. Res.*, **59**, 231 (1992).
18. Larry Hardesty, Amazon, MIT team up to add driver know-how to delivery-routing models, *Amazon Science Blog, https://www.amazon.science/blog/amazon-mit-team-up-to-add-driver-know-how-to-delivery-routing-models* (2021).