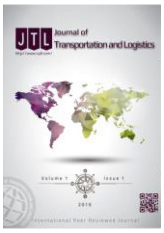

Available online at www.iujtl.com

JTL

Journal of Transportation and Logistics
7 (2) 2022

İSTANBUL UNIVERSITY PRESS

DOI: 10.26650/JTL.2022.1096085

RESEARCH ARTICLE

A Sequential Workforce Scheduling and Routing Problem for the Retail Industry: A Case Study

Uğur Eliiyi[1], Onur Uğurlu[2], Sel Özcan Tatari[3]

ABSTRACT

This study investigated the operational workforce scheduling and routing problem of a leading international retail company. Currently, the company plans to launch a new product into the Turkish market, which will be used in all its retail stores across the country. For the best marketing outcome, branding of all retail stores needs to be renewed by an outsourced workforce with a minimum of cost and time. We framed this as a workforce scheduling and routing optimization problem. Therefore, a two-stage solution was proposed. The retail stores were partitioned into disjoint regions in the first stage, and the schedules were optimized in the second stage. We employed the k-means clustering algorithm for constructing these regions. Two different heuristic approaches were applied to solve regional scheduling in the second stage of the algorithm since the resulting scheduling problem is NP-hard. Finally, a computational analysis was performed with real data and the results are discussed.

Keywords: Workforce Scheduling and Routing, Integer Programming, Heuristics

Submitted: 30.03.2022 • **Revision Requested:** 04.07.2021 • **Last Revision Received:** 12.10.2022 • **Accepted:** 20.10.2022

[1] **Corresponding author:** Uğur Eliiyi (Asst. Prof. Dr.), Izmir Bakırcay University, Department of Business, Izmir, Turkiye. E-mail: ugur.eliiyi@bakircay.edu.tr ORCID: 0000-0002-5584-891X

[2] Onur Uğurlu (Asst. Prof. Dr.), Izmir Bakırcay University, Department of Fundamental Sciences, Izmir, Turkiye. E-mail: onur.ugurlu@bakircay.edu.tr ORCID: 0000-0003-2743-5939

[3] Sel Özcan Tatari (Asst. Prof. Dr.), Izmir Bakırcay University, Department of Industrial Engineering, Izmir, Turkiye. E-mail: selozcan85@gmail.com ORCID: 0000-0002-4711-6663

Citation: Eliiyi, U., Ugurlu, O., & Ozcan Tatari, S. (2022). A Sequential workforce scheduling and routing problem for the retail industry: A case study. Journal of Transportation and Logistics. *Journal of Transportation and Logistics, 7*(2), 479-498. https://doi.org/10.26650/JTL.2022.1096085

This work is licensed under Creative Commons Attribution-NonCommercial 4.0 International License

1. Introduction

The retail industry covers a broad spectrum of sectors. Supermarkets, food and grocery stores, the apparel industry, consumer electronics, and gas stations are the major players in the retail market. Retailers are operated in a fragmented need environment with hybrid consumption, changing channels, big data, and real-time decisions (Berkhout, 2015). Due to fierce competition in today's global retail industry, companies started to invest more in the branding of their products. Hence, what motivates the users to return and remain with a specific retailer is a constant challenge for retailers. Brand recognition, therefore, is crucial for luring new customers and retaining existing ones. A brand renewal or re-branding process can sometimes be necessary for increasing the response of the target audience or for promoting new products. However, this process can take considerable time especially if physical re-branding is involved. In such a case, one challenge may be to complete all brand renewal operations in a timely manner to avoid any confusion while keeping the brand message consistent. To overcome this challenge, a previously planned brand renewal schedule should be developed, which will render a smooth transition.

This study was motivated by one of the leading international retail companies, which is continuously expanding and currently has around 750 stores all over Türkiye. The company has introduced a new product into the Turkish market, and all its retail stores needed to update their branding to include it. The company uses subcontractors to manage this process. As the branding of any store requires changes, a multi-stage decision process is involved. In the first step, a discovery operation for the corresponding store is conducted. Afterwards, the production of the set of new branding banners is performed according to the store size and specifications. At the last stage, which is renewal, the old branding materials are rolled out and replaced with new ones, as desired by the company. This branding renewal process is repeated every time a new product is launched into the market or for other reasons. Therefore, obtaining fast and effective solutions is crucial for the company decision-makers.

In this context, we considered the operational workforce scheduling and routing problem for this international company. The joint problem of scheduling and routing is managed within a two-stage optimization framework. While the first problem is to balance the total operational workload among several teams, the second problem focuses on obtaining a permutation of the stores visited by each team. The objective of the second problem is to minimize the total distance traveled by each team. After the new store regions are created, the second problem is reduced to the well-known Traveling Salesman Problem (TSP), which is NP-hard. Note that the integrated problem mentioned above can also be defined as the Multi-Depot, Multiple Traveling Salesman Problem (mDmTSP). The mDmTSP involves the decision of the set of nodes visited by each salesperson in non-conflicting schedules, where each salesperson, i.e., team, is located at different depots. Here, non-conflicting means that no salesperson is allowed to re-visit a node.

The main contribution of this study lies in providing an efficient solution to an existing practical problem of a large company. The optimization problem encountered by the company is an important operational one and the solutions provided can result in considerable savings in operational costs. Our second contribution is that the easy-to-use framework and efficient solutions provided to this important operational problem can be adapted and used for other similar applications by different companies or even in different sectors where worker groups are to be scheduled and routed separately. Although a two-stage framework can be regarded as one of the most common approaches in solving similar problems, our contribution lies in the employment of novel heuristics within this framework, which can be used by both researchers and practitioners for solving instances of the traveling salesman problem.

The rest of this paper is organized as follows: In Section 2, the operational workforce scheduling and routing problem on hand is introduced along with its problem-specific characteristics and related literature. The solution framework, including the clustering algorithm, the mathematical model and two heuristics for obtaining good-quality solutions for the problem are presented in Section 3. We report the results of our computational study in Section 4 along with managerial implications. Finally, Section 5 includes our conclusions and future work ideas.

2. Problem Context and Related Literature

This study is motivated by a real-life problem that is common in the retail industry. One of the leading international retail companies plans to introduce a new product into its domestic market in Türkiye. The company currently operates nearly 750 stores that are dispersed all over the country. The company plans to renew the branding of every one of these stores. Although the renewal process is outsourced to a third party, the whole rebranding is to be planned and coordinated by the retail company and completed as soon as possible; hence the time and cost efficiency of the branding operations are important.

The branding renewal includes a three-stage process; an assessment of the stores, store-specific production of the new branding, and a renewal of the old branding with the new one. The assessment is a discovery stage before the branding production, as it includes the evaluation and appraisal of each store that needs to be renewed. The company assigns a team to each store for the assessment process. The branding production requires a fixed time, independent of the store visited. This retail company can produce large amounts of new branding materials in a short time, so the operation times for manufacturing the new branding materials is of no importance. All three stages of the branding renewal operation are repeated every time a new product is launched, or when a brand “face lift” is required. The assessment stage must also be repeated, since the company is continuously expanding within the domestic market. New product launches can occur multiple times a year, so the frequency of rebranding is highly dependent on the market competition.

The renewal process includes rebranding materials, which is assumed to be readily available at the time of renewal. Note that both the assessment and renewal processes require on-site work, and there is a predetermined number of independent teams dedicated to these stages. Moreover, an additional problem constraint is that the decision-makers at the company require each team to visit at least three or more stores in one tour to justify their travel in terms of effort and time. Therefore, no team is allowed to visit just one or two stores in their working period.

With such a restrictive environment, we focused on optimization of worker allocation and scheduling for the assessment and renewal processes for the retail company. The problem involved making operational decisions as to which stores should be renewed using which worker team, and in what order. To address this problem, a scheduling and routing plan must be developed for each team so that all duties are covered. These scheduling and routing coordination make them an exceedingly difficult optimization problem.

The Workforce Scheduling and Routing Problem (WSRP) is a combination of personnel scheduling and vehicle routing which are both NP-Hard (Algethami & Landa-Silva, 2015). The WSRP studies encountered in the literature have applications in both the service and manufacturing industries. The assignment and scheduling of a workforce that served a regional location such as home health care (Mankowska et al., 2014), cargo collection and distribution (Liu et al., 2019), scheduling of technicians (Kovacs et al., 2012), security personnel routing and rostering (Misir et al., 2011), aircraft ground service scheduling (Ip et al., 2013), ship routing and scheduling (Pratap et al., 2019) are some of the problems. The existing studies include a vast range of modeling and solution approaches that were tailored according to a variety of applications. Interested readers can refer to the comprehensive survey by Castillo-Salazar et al. (2016), which reviewed workforce scheduling and routing studies.

In literature, the major interest in workforce scheduling and routing problems was either solely on balancing of the workload among workers, or the scheduling of the workforce (Castillo-Salazar et al, 2016). The proposed solution approaches naturally satisfy the workload balance among teams, so the focus of our literature review was on the routing part of the problem. In general, the routing in WSRP can be modeled as a Vehicle Routing Problem (VRP). However, in this study the problem can be modeled as Multiple Traveling Salesman Problem since there was no constraint on the use of vehicles.

The Multiple Traveling Salesman Problem (mTSP), which is a straightforward extension of the TSP, requires more than one salesperson to be used to cover the whole set of customers or nodes to be visited. The problem is of high practical importance. Derived from the well-known TSP, mTSP is more difficult. This is because the problem additionally requires the determination of the optimal allocation of nodes to the subtours of salespeople, such that each node is visited only once by one salesperson, and the total travel distance is minimized. The mTSP gained less interest in the literature than the well-known TSP, although it is more suited for scheduling and routing problems encountered in real-life. An excellent review of these studies was performed by Bektas (2006). An interesting study including several integer programming formulations of the mTSPs was also proposed by Kara and Bektas (2006).

Most of the studies in the literature focused on solving the mTSP by reducing it to the ordinary TSP, as the original problem is exceedingly difficult to solve. Venkateswara Reddy et al. (2010) addressed a method for transforming mTSP to TSP using Balanced centroids *k*-means clustering. Necula et al. (2018) proposed a min-max formulation for the mTSP with the objective of minimizing the maximum subtour of a salesperson. On the other hand, Chandran et al. (2006) studied the problem with the objective of balancing the workload among salespeople. They proposed a clustering approach for solving this problem.

As an example to non-linear formulations for the problem, Gilbert and Hofstra (1992) developed a non-linear model, as well as a heuristic approach to solve the multi-period mTSP with an application to a scheduling problem. Most researchers in the literature tackled the problem using problem-specific heuristics or metaheuristics, as it is well-known to be NP-hard. As a good example for these large number of studies, Masutti and De Castro (2008) presented an artificial neural network-based clustering approach for the mTSP. Bredström and Rönnqvist (2008) presented a mathematical programming model for the combined vehicle routing and scheduling problem with time windows and additional temporal constraints. In a later study, Allaoua et. al. (2013) proposed a matheuristic algorithm for the problem in a home health care context. The authors also reduced the Multi-Depot Traveling Salesman Problem to regular TSP for constructing the routes. The most well-known tour construction heuristics in the literature are the nearest neighbor, intersection, and greedy algorithms (Johnson & McGeoch, 1997; Davendra, 2010).

For routing optimization problems, decomposition-based matheuristics such as relax-and-fix, fix-and-optimize, and relax-fix-optimize have been used in literature. These algorithms were shown to be effective and efficient for routing-type problems (Friske et al., 2022). The relax-and-fit idea is based on breaking a mixed-integer linear problem into smaller subproblems and solving them iteratively to construct an initial solution. An example study by Uggen et al. (2013) developed an extended relax-and-fit matheuristic for a rich liquefied natural gas inventory routing problem and obtained good results. They used a time-based decomposition for reducing the original mathematical model into smaller subproblems. Fix-and-optimize algorithms were employed for routing problems by Goel et al. (2012) and Song and Furman (2013), as well. The appeal of employing these types of matheuristics lies in their ease of adaptation as well as their good performance for routing and TSP problems.

In our study, we employed a two-stage solution framework for a real-life problem. In line with the most common approach in literature, we first reduced our mTSP problem into several independent TSP problems via clustering and then treated each TSP separately. Our study is similar to the study by Chandran et al. (2006) in this respect, however, the proposed solution procedure of Chandran et al. (2006) did not minimize the distance travelled since it was designed to balance only the workload. For clustering of the worker groups, we used the well-known and well-adapted *k*-means clustering algorithm due to its ease of use and high adaptability, as in Necula et al. (2018). We also developed and employed simple problem specific heuristics for solving the TSP component of our problem. The solution framework, along with its details is presented in the next section.

3. The Solution Framework

The aggregate workforce scheduling and routing problem defined in the previous sections can be treated as an mDmTSP. However, as even the mTSP problem is NP-complete, the optimal solution of the joint problem was considered highly impractical for the real-life problem in this study. Instead, we decomposed the problem into several TSPs using clustering and in turn worked on independent TSPs for each resulting cluster. The aim of clustering is to balance the operational workload of the teams assigned to different regions and minimize the distance traveled as much as possible for each cluster. Hence, the first phase of our solution framework includes a solution for the workforce scheduling problem. After clustering, the second phase of our solution framework tries to obtain team tours with minimum distances traveled within each region/cluster. Hence, the workforce routing problem was decomposed into subproblems, each involving a TSP. For each TSP, we provided the decision makers with two fast and effective heuristic approaches to obtain feasible tours. For a better understanding, we illustrate our proposed solution framework in Figure 1.

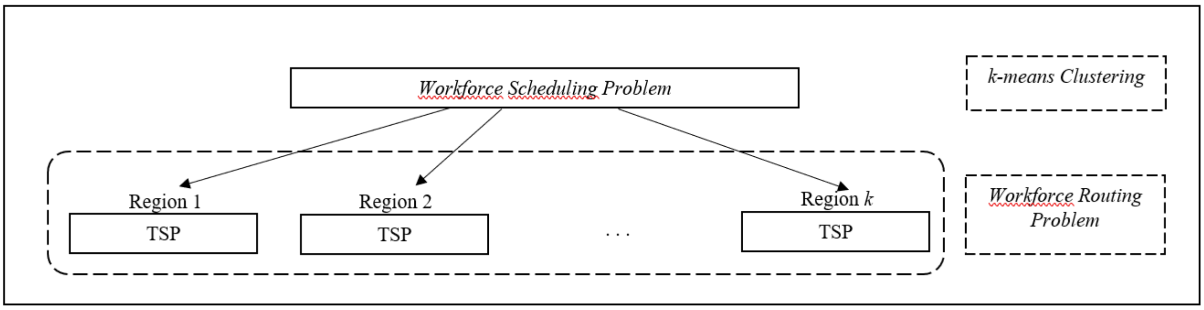

Figure 1. Proposed solution framework.

To decompose the main problem into several subproblems, we first partitioned the stores into a predetermined number of disjoint regions using the *k*-means clustering algorithm. This algorithm is one of the simplest and most efficient machine learning algorithms, developed by Lloyd (1982). This algorithm was employed in our solution framework for two reasons. First, the decision-makers in our study wanted the option to determine the number of resulting clusters and be able to try different scenarios easily to obtain different work plans. For this purpose, *k*-means algorithm provided a quick and flexible structure. The second reason was the ease of use, efficiency, and adaptability of this algorithm, which justifies its popularity among practitioners.

The algorithm aims to partition n points/nodes into k distinct clusters, in which each point belongs to the cluster with the nearest mean. The main steps of the *k*-means algorithm are as follows (MacKay, 2003):

1. Specify the number of clusters (k) and initialize k centroids randomly.
2. Assign each point to the closest centroid.
3. Update the cluster centroids as averages of the points contained within.
4. Repeat this procedure until there is no change to the centroids.

An example of stores grouped into clusters are visualized in Figure 2 on the country's map. As it can be seen in the figure, once the *k*-means clustering algorithm was applied to the mDmTSP, *k* independent TSPs were obtained. Five clusters were selected as an example in Figure 2.

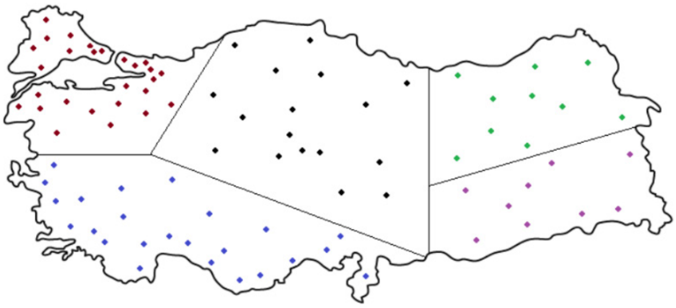

Figure 2. Example with five clusters (k=5) on Türkiye's map.

To solve the TSPs within each region after clustering, two different heuristic approaches are employed within the solution framework. The first heuristic includes a relax-fix-optimize (RFO) type matheuristic approach, which combines construction and improvement ideas to solve the resulting TSP. The general flow indicating the steps of the *RFO Matheuristic* is presented next.

3.1. The RFO Matheuristic

The proposed *RFO matheuristic* works in three stages:

- Relaxing the subtour elimination constraints,
- Enforcing a few subtours to be active,
- Elimination of the subtours.

The general flow diagram indicating the steps of the simple *RFO matheuristic* is provided in Figure 3. As illustrated in the figure, we first solved the TSP addressed by the mathematical model expressed by equations (1) through (5) for each region while excluding the subtour elimination constraints stated in (4).

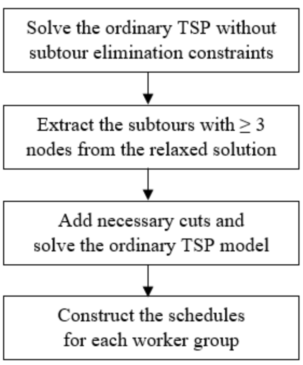

Figure 3. The relax-fix-optimize (RFO) matheuristic.

The definitions necessary for the formulated TSP and the mathematical model employed in the RFO matheuristic are as follows.

Sets, Indices, and Parameters of the TSP Model (RFO):

V: set of nodes (stores), i,j=1,2,…

E: set of edges

c_{ij} : distance between nodes i and j

Decision Variables of the TSP Model (RFO):

$$X_{ij} = \begin{cases} 1, & \text{if edge } (i,j) \text{ is in the optimal solution} \\ 0, & \text{otherwise} \end{cases} \qquad (i,j) \in E$$

Model TSP(RFO):

$$min \sum_{(i,j)\in E} c_{ij} X_{ij} \tag{1}$$

s. t.

$$\sum_{i\in V} X_{ij} = 1 , \qquad j \in V \tag{2}$$

$$\sum_{j\in V} X_{ij} = 1 , \qquad i \in V \tag{3}$$

$$\sum_{(i,j)\in E} X_{ij} \le |S| - 1 , \text{ for every proper subset } S \text{ of } V \tag{4}$$

$$X_{ij} \in \{0,1\}, \qquad (i,j) \in E \tag{5}$$

Once the constraint set (4) is excluded from the above formulation of TSP, the problem is reduced to the regular well-known *assignment problem*. The optimal solution of the assignment problem can be obtained in polynomial time; however, it may include several subtours consisting of two, three, or more stores in the resulting optimal solution.

Since the original problem required each salesperson to visit at least three or more stores in one tour, our heuristic only sought subtours with three or more nodes, i.e., independent subtours with more than two stores. Hence, once the optimal solution for the assignment problem was obtained, subtours containing 3 or more stores (if any) were recorded. Then, necessary cuts were generated as subtour preservation constraints pertaining to the identified subtours of desired length. In this manner, we ensured that these edges were preserved in the resulting TSP solution. After a set of such decision variables were fixed, the relaxed decision variables were optimized by solving the ordinary TSP(RFO) model once again. Note that, once subtours with more than or equal to three nodes were extracted from the resulting solution, a decision had to be made about omitting an edge for each subtour. For this purpose, the edge with the largest traveling distance was excluded by the heuristic for each subtour.

For a better understanding of the proposed *RFO matheuristic*, we provided a simple narrative example. Assume that after clustering, cluster 1 includes 20 stores. For this cluster, we relax the subtour elimination constraints (by omitting constraint set (4)) in the TSP(RFO) model and solve the resulting assignment problem to optimality. Assume that the subtours in Figure 4 appear in the optimal solution of the assignment problem.

Figure 4. Two subtours, each with three nodes and three edges.

The subtours in Figure 4 are 1-3-5-1 and 2-4-6-2. Now, assume that edges 5→1 and 6→2 have the maximum distances in each subtour. Hence, these edges are to be broken to eliminate these two subtours, while preserving the rest of the permutations with desired length. Accordingly, the following cuts $x_{13}=1, x_{35}=1, x_{24}=1, x_{46}=1$ are added to the ordinary TSP formulation for preserving the partial triplets added through these cuts (Figure 5).

Figure 5. Partial triplets, each with three nodes and two edges.

Subsequently, the solution of the ordinary TSP provides a single tour that contains the preserved partial triplets in the single tour (see Figure 6).

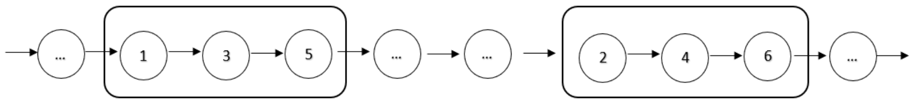

Figure 6. Feasible schedule after fixing partial triplets and optimizing the remaining TSP.

Our second heuristic, called the Two-Way Greedy (TWG) Heuristic, is a constructive effective heuristic that works fast for the TSP, and is presented in the next subsection.

3.2. The Two-Way Greedy (TWG) Heuristic

The proposed TWG heuristic is based on the greedy constructive approach for TSP, which starts with an empty solution and adds an edge to the solution according to some greedy criterion in an iterative manner. This process continues until a tour including all nodes is obtained. Unlike the ordinary greedy approach, our proposed heuristic has three stages: inclusion, removal, and merge. In the first stage, the algorithm starts with an empty solution and constructs an initial graph G^* from scratch by the greedy inclusion based on edge costs/lengths, starting with the shortest edge. In the second stage, the heuristic removes the redundant edges from G^* and creates disconnected paths. While doing so, the edges are sorted in descending order of their costs/lengths, hence the algorithm favors the removal of longer edges from the graph. Finally, a complete tour is constructed by merging the disjoint paths with the minimum cost/length obtained in the second stage to obtain a complete TSP tour. The proposed algorithm was originally developed for the multiple traveling salesman problem and presented in Nuriyev et al. (2018). It was shown that the algorithm found better results than the existing tour construction algorithms (Ugurlu, 2018). The steps of the algorithm are provided below:

Two Way Greedy (TWG) Heuristic:

Begin
    $G^* \longleftarrow 0$, degree() $\longleftarrow 0$
    Sort all edges in E in ascending order
    for all e_{ij} in E **do**
        if (degree(i) < 2 or degree(j) < 2) **then**
        include e_{ij} in G^*
        update degree(i) and degree(j)
    end if
    end for
    Sort all edges in G^* in descending order
    for all e_{ij} in G^* **do**
        if (degree(i) > 2 or degree(j) > 2) **then**
        remove eij from G^*
        update degree(i) and degree(j)
        end if
    end for
    Merge the paths/circles in G^* with minimum cost
End.

In the above pseudocode we can see the developed TWG algorithm has similar components with the traditional greedy algorithm for TSP. The traditional greedy algorithm gradually constructs a tour by repeatedly selecting the shortest edge and adding it to the tour if it does not increase the degree of any node by more than two (and thereby creating a subtour). However, in the inclusion stage of the TWG algorithm, an edge can be selected and added to the solution unless it increases the degree of both of its endpoints by more than two. Hence, TWG allows subtours in its inclusion stage. The difference of TWG can be better observed in the example provided in Figure 7.

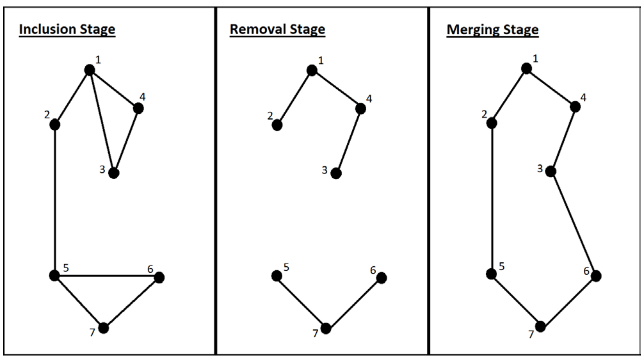

Figure 7. Main stages of the TWG algorithm.

Unlike a traditional greedy algorithm, the edges between nodes 1 and 3 and nodes 2 and 5, i.e., e_{13} and e_{25} can be added to the initial solution, thereby creating two subtours in Figure 7. This would not be possible through a traditional greedy heuristic, as it never allows subtours. After such an initial graph G^* is constructed in the inclusion stage, the TWG heuristic creates disconnected paths/tours by removing e_{13}, e_{25} and e_{56}, and obtains disconnected paths at the end of the removal stage. Finally, at the merging stage of the heuristic, a complete tour is obtained by merging disjoint paths by adding e_{25} and e_{36} to the final solution.

4. Computational Results and Discussion

The computational experiment in this study was conducted on an extensive set of real-life test problems. The optimal solutions for all mathematical models were obtained using the commercial optimization software IBM ILOG CPLEX Optimization Studio 12.9.0 on a computer with i5 processor, 8250 CPU and 8 GB RAM.

We first employed the *k*-means clustering algorithm for several *k* values to solve the workforce scheduling problem and balance the total number of stores over the *k* regions. The reason for obtaining clusters with different *k* values was one of the requirements of the decision-makers, as the number of worker groups was a determinant of operational cost. As a result, the number of distinct TSPs to be optimized increased. When considering that this problem must be solved every time there is a rebranding, the need for quick and effective solutions are better understood.

Euclidian distances are used to generate the clusters on a map of Türkiye. The real-life problem included 718 stores, dispersed around the country in a nonuniform fashion. Some store locations are remote, and the worker groups assigned to such locations needed to stay overnight to complete their operations. The branding renewal of all stores needed to be completed as early as possible. Moreover, as the company used subcontractors for this process; we assumed that the number of teams in the available workforce could vary within a predetermined range, each with identical capabilities. Due to the operational restrictions posed by the company regarding workforce efficiency and management, only six to ten clusters were included in the experimentation. Therefore, the *k*-means algorithm was run with parameter values of $k = 6,...,10$, and the stores were partitioned into $k = 6,...,10$ clusters as a result (see Table 1). Note that there was a trade-off between the total cost of the subcontracted teams and the time of completion of all operations. In other words, as the number of clusters, i.e., the number of teams increased, the total cost of the branding operation increased. However, the completion time of the branding operations decreased as a result.

The characteristics of the data set used for the workforce scheduling problem are reported in Table 1, where the second column indicates the total of Sums of Squared Errors (SSEs) within each cluster. The SSE is defined as the sum of the squared distances of each store to its closest centroid in *k*-means clustering. As seen on Table 1, the clustering effect in terms of reducing the within-cluster variance begins to appear after using k=8 partitions. As the number of clusters increases to 9 and 10, total SSE values decrease as clustering alternatives appear for the case problem locations. We also reported the number of stores within each cluster in the third column. The last two columns respectively report the mean and the standard deviation of the corresponding distance matrices.

Table 1: Characteristics of the tested problem clusters.

Clusters partitioned with *k*	SSE	Cluster number	Number of Stores	Mean *(km)*	Standard deviation *(km)*
k=6	44166	Cluster 1	116	199	137.04
		Cluster 2	44	280	197.54
		Cluster 3	74	252	160.93
		Cluster 4	199	115	94
		Cluster 5	123	239	159
		Cluster 6	162	154	95
k=7	44835	Cluster 1	76	122	84.79
		Cluster 2	47	291	182.09
		Cluster 3	75	142	104.59
		Cluster 4	191	122	107.22
		Cluster 5	111	119	93.91
		Cluster 6	110	186	110.56
		Cluster 7	108	199	144.97
k=8	42948	Cluster 1	37	232	171.39
		Cluster 2	30	213	127.04
		Cluster 3	73	146	113.12
		Cluster 4	188	113	95.48
		Cluster 5	111	119	93.91
		Cluster 6	108	182	108.42
		Cluster 7	95	150	99.34
		Cluster 8	76	122	84.79
k=9	26481	Cluster 1	37	232	171.39
		Cluster 2	30	213	127.04
		Cluster 3	73	146	113.12
		Cluster 4	43	72	43.08
		Cluster 5	111	119	93.91
		Cluster 6	108	182	108.42
		Cluster 7	95	150	99.34
		Cluster 8	146	87	84.35
		Cluster 9	75	117.11	83.73
k=10	14062	Cluster 1	37	232	171.39
		Cluster 2	30	213	127.04
		Cluster 3	74	146	113.12
		Cluster 4	43	72	43.08
		Cluster 5	118	138	100.84
		Cluster 6	87	181	115.71
		Cluster 7	94	147	96.56
		Cluster 8	146	86.02	84.35
		Cluster 9	64	98	74.15
		Cluster 10	25	76	44.84

As stated earlier, balancing the total operational workload among several teams is of concern in the first phase of our solution framework. Based on this consideration, it is possible that the number of stores in the clusters formed at this phase could be close to one another. However, it can be seen in Table 1 that there is still a variability in cluster store numbers. This variability is because the stores are dispersed over the country in a nonuniform fashion. Some stores are remote and considerable distances need to be traveled to get to those locations. Once a worker team completes a shift on a remote location, the team has to stay overnight, as well. Therefore, the balancing of the workload is not about the total number of stores assigned to teams, but the total tour length (i.e., the objective function value of the TSP). This approach not only minimizes the total length of travel among teams, but also indirectly minimizes the makespan, i.e., the time needed to complete the last branding operation.

We tested the performance of two heuristics, RFO and TWG, and compared them with the CPLEX TSP solution with a 600 second time limit. As RFO is an integer programming-based matheuristic, we limited the computational run time of the algorithm with 600 seconds, as well. As stated earlier, the company management desired fast and effective solutions, which could be obtained repeatedly for different sized clusters. This requirement called for the need to obtain time-limited solutions. Initial pilot runs included a 120-second time limit as well, but as the results within this time limit were not satisfactory, they are not included. CPLEX could not produce feasible solutions for many of the problem instances within this period, and heuristic performance was inferior when compared with the 600-second results. A 3600-second time limit was also assessed in the pilot experimentation. However, CPLEX resulted in memory overruns for TSP instances having over 100 stores within this time limit, whereas there was little improvement over the 600-second results for smaller instances. As instances having over 100 stores occur in every cluster size in Table 1, complete solutions for the whole problem could not be obtained within this time limit.

Tables 2 and 3 summarize the optimality gap percentages and the total distances of the two approaches within the 600-second time limit, the TSP integer programming formulation and the RFO heuristic. The optimality gaps are computed for each algorithm independently. Namely, the optimality gap values under the TSP-600 column represent the percentage gaps between the best lower bound and best upper bound obtained by CPLEX within the time limit. On the other hand, the optimality gaps reported under the RFO-600 column are those obtained from the solutions of the TSP(RFO) model. Hence, the optimality gaps are not comparable among these two columns. As TWG performs superbly in terms of time and obtains solutions in less than one second for every instance, we reported the computation times and the tour lengths for each cluster for this heuristic.

In Tables 2 and 3, the first lines of the clusters represent the optimality gaps for the TSP and RFO, and the computational time for TWG in seconds. The second lines of the clusters represent the tour lengths (km) found by the TSP, RFO and TWG, respectively. The bold figures in the tables indicate the minimum objective function value for each cluster. Note that the objective function values, i.e., the tour lengths are directly comparable among all three columns. In small instances, i.e., for cluster sizes with less than 80 nodes, a better performance by the TSP formulation was observed over the remaining approaches. In these instances, the optimality gaps for the time-limited CPLEX solutions are less than 10%.

Table 2: Computational Results for k=6 to 8.

Clusters partitioned with k	Cluster number	TSP-600	RFO-600	TWG
k=6	Cluster 1	48.89%	25.19%	*(0.075)*
		3373	2489	**2332**
	Cluster 2	0.57%	0.00%	*(0.03)*
		2122	2374	2166
	Cluster 3	14.92%	3.29%	*(0.018)*
		2567	**2540**	2602
	Cluster 4	72.15%	10.83%	*(0.557)*
		7033	2405	**2220**
	Cluster 5	40.56%	20.96%	*(0.087)*
		2980	2384	**2091**
	Cluster 6	33.17%	0.00%	*(0.024)*
		3367	8000	**2641**
k=7	Cluster 1	12.38%	0.00%	*(0.016)*
		1212	2853	**1205**
	Cluster 2	3.00%	0.64%	*(0.004)*
		2384	5257	2454
	Cluster 3	1.98%	1.94%	*(0.017)*
		1598	1615	1644
	Cluster 4	47.15%	33.33%	*(0.476)*
		3148	2721	**1973**
	Cluster 5	15.92%	9.70%	*(0.063)*
		1756	1762	**1662**
	Cluster 6	4.45%	2.97%	*(0.056)*
		2000	2103	**1982**
	Cluster 7	30.21%	0.12%	*(0.065)*
		2750	7954	**2330**
k=8	Cluster 1	6.37%	4.91%	*(0.001)*
		1773	2116	1791
	Cluster 2	0.00%	0.00%	*(0.001)*
		1703	1796	1706
	Cluster 3	4.98%	2.61%	*(0.016)*
		1605	1619	1663
	Cluster 4	32.68%	19.65%	*(0.439)*
		2269	2078	**1840**
	Cluster 5	16.26%	8.00%	*(0.062)*
		1763	1730	**1662**
	Cluster 6	17.14%	2.77%	*(0.055)*
		2112	1998	**1994**
	Cluster 7	16.97%	10.78%	*(0.044)*
		1964	1911	**1855**
	Cluster 8	10.05%	5.10%	*(0.016)*
		1179	1204	1205

Although TSP-600 performs better for small-size clusters, in Tables 2 and 3 it shows that the TWG heuristic also performs very well for these instances, producing close results to those obtained by CPLEX. For example, for cluster 2 of k=6, TSP-600 obtained a tour length of 2122, which is approximately optimal. On the other hand, TWG obtained a tour length of 2166 in 0.03 seconds, which is just 2% longer. This comparison is discussed further through Table 4. In larger instances, i.e., clusters with more than 80 nodes, TWG clearly outperformed the remaining two solution approaches in terms of solution quality and computation time, as the underlying complexity of the problem prevails with larger node sizes.

One interesting result was that small optimality gaps could be observed for the RFO heuristic in many of the real-life instances, although the objective function values were inferior when compared with the other two methods. Although the optimality gap percentages for the RFO were lower than time limited CPLEX solutions in most instances, there were only two clusters where RFO yielded the best objective function value. An example can be observed in Table 2, for k=6, cluster 6. Although the optimality gap for RFO is zero for this instance, the objective function value of 8000 obtained with this heuristic is clearly dominated by the other two approaches. The best solution was obtained by TWG with an objective function value of 2641. The reason for this apparent conflict was that the added cuts by RFO resulted in the exclusion of the global optimal solution from the search space, and the algorithm was stuck at a local minimum for these problem instances. However, despite this critical observation, the overall performance of the algorithm was remarkable. The objective function values obtained with this heuristic were very close to those produced by TWG in most cases.

Table 3: Computational Results for *k*=9 and 10.

Clusters partitioned with *k*	**Cluster number**	**TSP-600**	**RFO-600**	**TWG**
k=9	Cluster 1	6.32%	4.98%	*(0.001)*
		1773	2116	1791
	Cluster 2	0.00%	0.00%	*(0.001)*
		1703	1796	1706
	Cluster 3	4.91%	2.62%	*(0.016)*
		1604	1619	1663
	Cluster 4	0.00%	0.00%	*(0.002)*
		753	801	792
	Cluster 5	19.77%	8.87%	*(0.062)*
		1840	1747	**1662**
	Cluster 6	16.41%	2.70%	*(0.055)*
		2094	1998	**1994**
	Cluster 7	20.30%	12.48%	*(0.044)*
		2046	1946	**1855**
	Cluster 8	47.27%	26.80%	*(0.175)*
		1735	1393	**1135**
	Cluster 9	11.03%	0.00%	*(0.015)*
		1165	8000	1169
k=10	Cluster 1	6.31%	5.06%	*(0.001)*
		1773	2116	1791
	Cluster 2	0.00%	0.00%	*(0.001)*
		1703	1796	1706
	Cluster 3	4.97%	2.61%	*(0.016)*
		1605	1619	1663
	Cluster 4	0.00%	0.00%	*(0.002)*
		753	801	792
	Cluster 5	31.21%	0.00%	*(0.08)*
		2004	1674	**1594**
	Cluster 6	9.20%	1.82%	*(0.027)*
		1678	1688	1695
	Cluster 7	32.39%	5.95%	*(0.036)*
		2198	**1741**	1844
	Cluster 8	46.42%	27.47%	*(0.175)*
		1710	1401	1135
	Cluster 9	8.82%	0.00%	*(0.009)*
		1013	1086	1015
	Cluster 10	0.00%	0.00%	*(0.001)*
		669	699	673

Table 4 summarizes the cluster-based improvement percentages of the TWG heuristic over the second-best approach, i.e., either the time-limited CPLEX or the RFO heuristic. The positive improvement percentages clearly indicate that TWG outperformed the remaining two approaches in obtaining good quality solutions for the problem. In 17 instances out of 40, TWG yielded the best performance, and remarkable improvements (more than 10%) could be recorded in almost 5 instances.

Table 4: Cluster-based improvement percentages of the TWG algorithm.

Clusters partitioned with *k*	***k*=6**	***k*=7**	***k*=8**	***k*=9**	***k*=10**
Cluster 1	6.31%	0.58%	-1.02%	-1.02%	-1.02%
Cluster 2	-2.07%	-2.94%	-0.18%	-0.18%	-0.18%
Cluster 3	-2.44%	-2.88%	-3.61%	-3.68%	-3.61%
Cluster 4	7.69%	27.49%	11.45%	-5.18%	-5.18%
Cluster 5	12.29%	5.35%	3.93%	4.87%	4.78%
Cluster 6	21.56%	0.90%	0.20%	0.20%	-1.01%
Cluster 7	-	14.96%	2.93%	4.68%	-5.92%
Cluster 8	-	-	-2.21%	18.52%	18.99%
Cluster 9	-	-	-	-0.34%	-0.20%
Cluster 10	-	-	-	-	-0.60%
Average	7.22%	6.21%	1.44%	1.99%	0.61%
Total distance TSP-600 (km)	21,442	14,838	14,368	14,713	15,106
Total distance RFO-600 (km)	20,192	24,265	14,452	21,416	14,621
Total distance TWG (km)	14,052	13,250	13,716	13,767	13,908
Improvement in distance (%)	30.41%	10.70%	4.54%	6.43%	4.88%

When the average improvements of all clusters were examined for *k*=6 to 10, an important observation could be made. Although some of the improvements are negative throughout the table, meaning that TWG is outperformed for those individual cluster instances, it is never outperformed when averaged over all clusters. This means that the TWG algorithm produced close results to the best one when it performed worse, but it obtained remarkably better results for larger clusters, yielding an overall improvement in all cases. Hence, the TWG heuristic was clearly dominant over the other two approaches for practical use. While examining this table, it should be noted that the structure of the resulting clusters in different columns are different from one another, as reported in Table 1.

The improvement by TWG is better observed when the total distance obtained over all clusters by different algorithms are compared. In Table 4, the *Total distance* rows present the total tour lengths of all clusters for *k*=6 to 10 obtained by the three approaches, whereas the last row of the table presents the improvement percentage (in terms of total distance) of TWG over the second-best approach, i.e., either the time-limited CPLEX or the RFO heuristic. The total distances are obtained by adding all distances within each cluster, i.e., the sum of all tours. The *Improvement in distance (%)* obtained for *k*=6 is especially remarkable with a dramatic 30.41%. This means that the results obtained in extremely short computation times with TWG result in a much shorter total distance for the teams when compared to RFO, which is the second-best heuristic for *k*=6. Positive *Improvement in distance* values in all columns for TWG reveal the dominance of this heuristic when the whole problem is considered, both in solution quality and solution time.

Figure 8 also illustrates how the total distances traveled by all teams vary depending on the clusters generated. From the figure, a clear dominance of the TWG heuristic can be observed for every cluster. On the other hand, the RFO heuristic has large jumps, indicating that it is not robust enough for the real-life data set on hand. The large jumps are due to instances where the heuristic is stuck at local optima and result in huge objective function values for a cluster, thereby affecting the summation.

Note that the total distance traveled is minimized at 7 clusters for the best performing heuristic, i.e., TWG. This means that this specific case problem for retail stores in Türkiye should be managed in 7 different regions, which would yield the most efficient solution for the company. TWG obtained 10.70% improvement in total distance over the second-best result for this configuration of the problem, in just a fraction of a second. The decision makers were impressed by the result, both in terms of the total distance travelled and the smaller number of teams required as a result. This result optimized the trade-off that was mentioned at the start of this section to a certain extent. However, due to different management considerations, a greater or smaller number of clusters might be preferred.

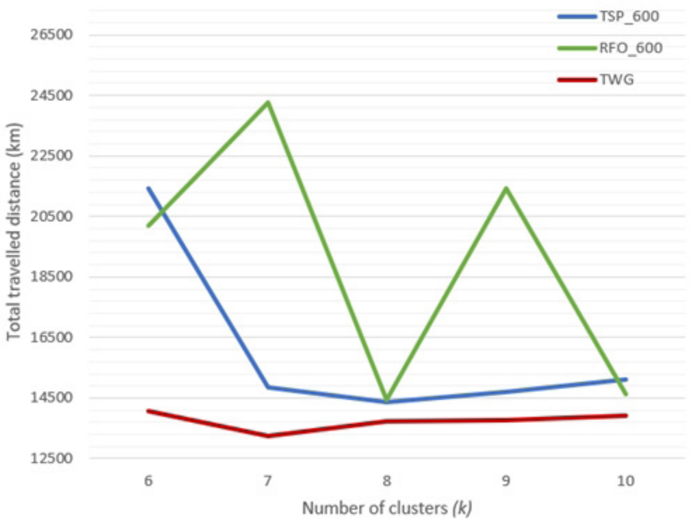

Figure 8. Performance of the solution approaches in terms of total distance traveled vs. the number of clusters.

5. Conclusion

In this study, we addressed an operational workforce scheduling and routing problem for an international retail company operating in Türkiye and provided a two-stage solution approach for this optimization problem. In the first stage, we balanced the operational workload among teams using the *k*-means clustering algorithm. Next, we focused on obtaining an efficient tour schedule for each team in the second stage, with an objective of minimizing the total travel distance. This was required for the goal of completing the whole branding in the fastest way possible. The aggregate problem is regarded as a multi-depot, multiple traveling salesman problem. Since even the multiple TSP is NP-complete, we employed simple and effective solution approaches to solve this practical real-life problem. We compared their performance with a regular TSP solution via a commercial solver within a practical time limit. We reported the percentage improvements on the solutions of the problems evaluated, as well as their optimality gaps.

The proposed solution framework achieved significant improvements in several instances, obtaining remarkable gains in the total distance traveled. The outcomes of computational experimentation showed remarkable gains to the managerial decision-makers. Our solution approaches may provide flexibility to companies for evaluating various scenarios depending on various cluster configurations.

The sensitivity analysis on the number of clusters indicated that the minimum distance traveled is not directly proportional to the number of clusters. Therefore, it is advisable for company management to consider all feasible alternative scenarios. As a future study, we plan to enrich our solution approach by employing more sophisticated heuristic approaches for the specific problem at hand. Another interesting extension may include developing a mixed integer programming formulation for deciding the daily schedules of each team by considering the daily working time windows.

Peer Review: Externally peer-reviewed.
Conflict of Interest: Author declared no conflict of interest.
Financial Disclosure: Author declared no financial support
Author Contributions: Conception/Design of Study- U.E., O.U., S.Ö.T.; Data Acquisition- U.E., O.U., S.Ö.T.; Data Analysis/Interpretation- U.E., O.U., S.Ö.T.; Drafting Manuscript- U.E., O.U., S.Ö.T.; Critical Revision of Manuscript- U.E., O.U., S.Ö.T.; Final Approval and Accountability- U.E., O.U., S.Ö.T.

References

Algethami, H., & Landa-Silva, D. (2015). A study of genetic operators for the workforce scheduling and routing problem. In *Proceedings of the 11th Metaheuristics International Conference* (MIC 2015, pp. 1-11). Agadir; Morocco.

Allaoua, H., Borne, S., Létocart, L., & Calvo, R. W. (2013). A matheuristic approach for solving a home health care problem. *Electronic Notes in Discrete Mathematics*, *41*, 471-478. https://doi.org/10.1016/j.endm.2013.05.127

Bektas, T. (2006). The multiple traveling salesman problem: an overview of formulations and solution procedures. *Omega*, *34*(3), 209-219. https://doi.org/10.1016/j.omega.2004.10.004

Berkhout, C. (2015). *Retail marketing strategy: Delivering shopper delight*. Kogan Page Publishers, USA.

Bredström, D., & Rönnqvist, M. (2008). Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. *European Journal of Operational Research*, *191*(1), 19-31. https://doi.org/10.1016/j.ejor.2007.07.033

Castillo-Salazar, J. A., Landa-Silva, D., & Qu, R. (2016). Workforce scheduling and routing problems: literature survey and computational study. *Annals of Operations Research*, *239*(1), 39-67. https://doi.org/10.1007/s10479-014-1687-2

Chandran, N., Narendran, T. T., & Ganesh, K. (2006). A clustering approach to solve the multiple traveling salesmen problem. *International Journal of Industrial and Systems Engineering*, *1*(3), 372-387. https://doi.org/10.1504/IJISE.2006.009794

Davendra, D. (Ed.). (2010). *Traveling salesman problem: Theory and applications*. BoD–Books on Demand.

Friske, M. W., Buriol, L. S., & Camponogara, E. (2022). A relax-and-fix and fix-and-optimize algorithm for a Maritime Inventory Routing Problem. *Computers & Operations Research*, *137*, 105520. https://doi.org/10.1016/j.cor.2021.105520

Gilbert, K. C., & Hofstra, R. B. (1992). A new multiperiod multiple traveling salesman problem with heuristic and application to a scheduling problem. *Decision Sciences*, *23*(1), 250-259. https://doi.org/10.1111/j.1540-5915.1992.tb00387.x

Goel, V., Furman, K. C., Song, J. H., & El-Bakry, A. S. (2012). Large neighborhood search for LNG inventory routing. *Journal of Heuristics*, *18*(6), 821-848. https://doi.org/10.1007/s10732-012-9206-6

Ip, W. H., Wang, D., & Cho, V. (2012). Aircraft ground service scheduling problems and their genetic algorithm with hybrid assignment and sequence encoding scheme. *IEEE Systems Journal*, *7*(4), 649-657. https://doi.org/10.1109/JSYST.2012.2196229

Johnson, D. S., & McGeoch, L. A. (2018). The traveling salesman problem: a case study. In E. Aarts & J. Lenstra (Ed.), *Local Search in Combinatorial Optimization* (pp. 215-310). Princeton: Princeton University Press, USA. https://doi.org/10.1515/9780691187563-011

Kara, I., & Bektas, T. (2006). Integer linear programming formulations of multiple salesman problems and its variations. *European Journal of Operational Research*, *174*(3), 1449-1458. https://doi.org/10.1016/j.ejor.2005.03.008

Kovacs, A. A., Parragh, S. N., Doerner, K. F., & Hartl, R. F. (2012). Adaptive large neighborhood search for service technician routing and scheduling problems. *Journal of Scheduling*, *15*(5), 579-600. https://doi.org/10.1007/s10951-011-0246-9

Liu, X., Luo, M., & Zhao, Y. (2019). Integrating route optimisation with vehicle and unloading dock scheduling in LCL cargo collection. *International Journal of Shipping and Transport Logistics*, *11*(2-3), 262-280. https://doi.org/10.1504/IJSTL.2019.099273

Lloyd, S. (1982). Least squares quantization in PCM. *IEEE Transactions on Information Theory*, *28*(2), 129-137. https://doi.org/10.1109/TIT.1982.1056489

MacKay, D. (2003). An example inference task: Clustering. *Information Theory, Inference and Learning Algorithms*, *20*, 284-292.

Mankowska, D. S., Meisel, F., & Bierwirth, C. (2014). The home health care routing and scheduling problem with interdependent services. *Health Care Management Science*, *17*(1), 15-30. https://doi.org/10.1007/s10729-013-9243-1

Masutti, T. A., & de Castro, L. N. (2008, July). A Clustering Approach Based on Artificial Neural Networks to Solve Routing Problems. In *2008 11th IEEE International Conference on Computational Science and Engineering* (pp. 285-292). IEEE. https://doi.org/10.1109/CSE.2008.58

Misir, M., Smet, P., Verbeeck, K., & Vanden Berghe, G. (2011). Security personnel routing and rostering: a hyper-heuristic approach. In *Proceedings of the 3rd International Conference on Applied Operational Research* (Vol. 3, pp. 193-205). Tadbir; Canada.

Necula, R., Raschip, M., & Breaban, M. (2018). Balancing the subtours for multiple TSP approached with ACS: Clustering-based approaches vs. MinMax formulation. In *EVOLVE-A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation VI* (pp. 210-223). Springer, Cham. https://doi.org/10.1007/978-3-319-69710-9_15

Nuriyev, U., Ugurlu, O., & Nuriyeva, F. (2018). A Simple Algorithm for The Multi-Depot Multiple Traveling Salesman Problem. In. *Proceedings of the 6th International Conference on Control and Optimization with Industrial Applications* (Vol. 2, pp. 235-237). Baku; Azerbaijan.

Pratap, S., Zhang, M., Shen, C. L., & Huang, G. Q. (2019). A multi-objective approach to analyse the effect of fuel consumption on ship routing and scheduling problem. *International Journal of Shipping and Transport Logistics*, *11*(2-3), 161-175. https://doi.org/10.1504/IJSTL.2019.099270

Song, J. H., & Furman, K. C. (2013). A maritime inventory routing problem: Practical approach. *Computers & Operations Research*, *40*(3), 657-665. https://doi.org/10.1016/j.cor.2010.10.031

Uggen, K. T., Fodstad, M., & Nørstebø, V. S. (2013). Using and extending fix-and-relax to solve maritime inventory routing problems. *TOP*, *21*(2), 355-377. https://doi.org/10.1007/s11750-011-0174-z

Ugurlu, O. (2018). Rotalama problemleri için algoritmik yaklaşımlar. Doktora Tezi, Ege Üniversitesi, Fen Bilimleri Enstitüsü, İzmir.

Venkateswara Reddy, P., Kumar, A. C. S., Bhat, M. S., Dhanalakshmi, R., & Parthiban, P. (2010). Balanced centroids (BC) k-means clustering algorithm to transform MTSP to TSP. *International Journal of Logistics Economics and Globalisation*, *2*(3), 187-197. https://doi.org/10.1504/IJLEG.2010.036299