Strategic Route Planning: An Adaptive BR-Heuristic for Multi-Depot Logistics

Marc Escoto[0009-0001-1879-3796], Antoni Guerrero[0000-0002-2118-9506], and Verónica Medina[0000-0002-3353-4434]

Research Center on Production Management and Engineering (CIGIP), Universitat Politècnica de València, 03801 Alcoy, Spain
{mescgom, aguepor1, vmedrid}@upv.es

Abstract. The multiple traveling salesman problem (mTSP) is an extension of the well-known traveling salesman problem (TSP), requiring multiple agents to visit a set of nodes and return to their respective depot. This problem finds wide-ranging applications in robotics, transportation and networking, among other domains. Moreover, it can be readily extended to a vehicle routing problem (VRP) by introducing extra constraints. This article presents a hybrid method for addressing the multi-depot closed path mTSP by integrating a bias-randomized heuristic with iterative local search (ILS). Furthermore, this algorithm is enhanced with a self-tuning mechanism, eliminating the need for time-consuming fine-tuning processes. The results provided by this algorithm with a different number of depots are analyzed.

Keywords: Multiple Traveling Salesman Problem · Bias-Randomization

1 Introduction

Imagine we have a fixed set of cities or locations that we need to visit, and it is essential to do so in the most efficient way possible. This problem, which can be stated as finding the shortest route that connects all of those places, is precisely what the TSP solves. Each location can only be visited once, and the salesman must return to the starting point. Note that this problem is often represented as a (di)graph, where each city or place corresponds to a node. Hence, we will commonly refer to each location as a node, while the starting point is known as the depot. The TSP has become one of the most studied problems since it is the basis of many scheduling and routing challenges, such as the VRP or the pick-up delivery problem. Each extension can be created generalizing the TSP, meaning that the NP-hard nature of the problem remains, and heuristics or other algorithms which produce near-optimal solutions are needed for their resolution. The primary objective of the TSP is to minimize a certain cost, whether it is time, distance, money, or a combination of some of them.

There are numerous variants of the TSP, one of which is the mTSP. In this scenario multiple salesmen are tasked with visiting all nodes, and the objective is to optimize routes, considering that all salesmen must start and finish their route at the depot. This extension of the TSP serves as the foundation for various related problems, such as the VRP. For instance, when considering the VRP as a last-mile delivery problem, a certain number of vehicles (salesmen) are tasked with delivering goods to customers' locations (nodes), with each vehicle beginning and ending its route at the depot. However, some scenarios require more flexibility, such as having multiple depots to minimize route costs and better accommodate customers' needs. In this case, we have the multi-depot mTSP (MDmTSP), where each node must be visited exactly by only one salesman, and each salesman starts and ends its route at their respective initial depot. This problem can also be referred to as the closed path MDmTSP, as there exists a variant known as the open path MDmTSP where it is not mandatory for salesmen to come back to their original depot, allowing them to stay at the last visited node or return to another depot. For more information, an extensive review of the generalized TSP and mTSP can be found at [12] and [3], respectively.

Finally, this paper introduces a bias-randomized heuristic with ILS and a self-tuning mechanism for the closed path MDmTSP. In Section 2, an overview of the related work is provided, showcasing the importance of the problem. In Section 3, the problem addressed in this article is formally introduced. Section 4 details the solving methodology, presenting a flow diagram of the proposed method followed by an in-depth explanation of each aspect of the algorithm. Section 5 delves into the results achieved by the algorithm across different problem instances and numbers of depots. Finally, in Section 6 the managerial insight of the method presented in this article is shown, while Section 7 offers a summary of the article and suggests future research directions and potential improvements.

2 Related Work

Despite the utility that developing an effective method for solving the mTSP would offer, relatively little research has been conducted in recent years, as other routing problems have gathered more attention from researchers. This trend is further accentuated by the fact that the mTSP with a single depot has received more focus than the multi-depot mTSP. In this section, related work regarding this problem is analyzed. Firstly, several conducted studies showcase the utility of mTSP in real-life applications.

In [6], an alternative for the organization of spraying missions in mountainous areas using several multicopters in collaboration is presented. The challenge is posed as a single-depot mTSP and three different algorithms are proposed to address it: the classical mTSP, the combined Cluster-TSP algorithm and the decoupled Cluster-TSP algorithm. Simulation results show that the classical mTSP algorithm allocates tasks equally, while the combined Grouping-TSP algorithm provides the optimal solution.

Another interesting application is raised in [13], where it focuses on the design of routes for several unmanned drones (UAVs) to collect information after a catastrophe by cooperative video inspection and uploading information to ground base stations. Three-dimensional routes for several UAVs are designed to minimize inspection time and maximize communication time for video upload. The solution of a multiple traveling salesman problem is proposed to design the drone trajectories and numerical results show that this design can significantly reduce the video inspection time, enabling fast response and efficient planning in case of catastrophic disaster.

Among the several approaches that have been taken to tackle the mTSP, a common method is the use of neural networks to extend a TSP into a mTSP. For instance, in [10] a genetic algorithm based on metaheuristics with tournament selection (GATS) is presented to solve a single-depot bi-objective mTSP (BmTSP) with the load-balancing constraint, where the first objective is to minimize the total travel distance and the second objective minimizes the total time. GATS integrates mixed strategies into the mutation operation and was compared with other genetic approaches on different TSPLIB-derived datasets. Experiments showed that the proposed GATS obtained efficient Pareto solutions of the BmTSP.

If we focus on sustainability, by reducing carbon emissions in transportation, a quasi-oppositional multi-objective Jaya algorithm (RPAL-based MOQO Jaya) is used in [1]. The mTSP is solved, considering multiple sellers and risk and carbon constraints. Real data from Surat city is used to observe the effects of seller selection in different models so the performance of the proposed algorithm is compared and its effectiveness in optimizing transportation routes is demonstrated.

When addressing the MDmTSP, various approaches have been taken. While most of them focus on finding a direct solution, [11] explored a different approach, in which the MDmTSP was converted into a single asymmetric TSP by incorporating a set of duplicate nodes, each corresponding to a depot vertex, therefore facilitating the management of the problem.

An example of a direct approach is illustrated in [5], where a method for tackling the mTSP was introduced. This method involved transforming a complex graph into a simpler one and solving the mTSP based on the simplified results. The simplified model was generated from an initial connected graph removing redundant edges, this is, edges which connect nodes with more than two edges connecting them to other nodes. Subsequently, m routes were generated from the connected graph by generating sub-graphs and finally a 2-opt was used to optimize the solution. Computational experiments were used to test the method and compare the results obtained with the optimal result for that instance with one salesman.

More recent approaches focus on the use of metaheuristics to develop solutions. [4] presented a memetic algorithm for addressing the minmax mTSP with single and multiple depots, by integrating an edge assembly crossover operator, a streamlined variable neighborhood search and a post-optimization technique and [9] proposed a heuristic that created a feasible solution using an ILS to improve the solutions found for the minmax heterogeneous MDmTSP, both obtaining very good results in their respective computational tests.

3 Problem Definition

Kara et al. [8] investigated various formulations for the mTSP. Their article presented mathematical models for the single-depot mTSP and several versions of the multi-depot mTSP. The method outlined in this article aims to address the closed-path MDmTSP, which is a specific case of the fixed-destination MmTSP presented by the authors. Therefore, the problem definition will adhere to the model described in [8].

The problem is described as a complete directed graph $G = (V, A)$, where A is a set of arcs and V is a set of nodes. The cost matrix $C = (c_{ij})$ represents the cost of traversing the arc $(i, j) \in A$. The nodes of the graph can represent depot nodes or customer nodes. We define $D \subseteq V$ as the set of depots and $V' = V \backslash D$ as the set of customer nodes. There is a total of m salesmen which will traverse the nodes, where m_i of them are located in depot $i \in D$ initially. The objective is to find tours for all salesmen, ensuring that each customer is visited exactly once, with each salesman visiting a number of customers between a lower limit K and an upper limit L. The term "fixed-destination" denotes the requirement for all salesmen to return to their original depot.

In the particular case studied in this paper, several considerations must be made. Firstly, all nodes are connected to each other, and the distances between nodes are symmetrical. Therefore, we could describe the problem with an undirected graph, resulting in a symmetrical cost matrix. Secondly, a single salesman is located at each depot, hence the number of depots is equal to the number of salesman, i.e. $|D| = m$, and the number of salesmen per depot, m_i, will be set to one, i.e. $m_i = 1 \ \forall i$. Finally, as the method presented ensures balanced routes regarding the number of nodes visited by each salesman, the number of visited nodes per salesman will be the number of customer nodes $|V'|$ divided by the number of salesman m. Given that this number is not ensured to be natural, the lower and upper limits for the nodes visited by each salesmen are determined by $\lfloor \frac{|V'|}{|m|} \rfloor$ and $\lfloor \frac{|V'|}{|m|} \rfloor + 1$, respectively.

Various objective functions can be considered for this problem, with the most commonly used ones being the 'minmax' and the 'minsum'. The minmax objective function seeks to minimize the longest route among those taken by the salesmen, while the minsum objective function aims to minimize the total distance covered by each salesman on their route. In our study, we consider the minsum objective function. Hence, it can be formulated as:

$$min \sum_{i \in D} d_i$$

where d_i represents the distance of the route taken by the salesman departing from depot i.

4 Solving Methodology

In this article, a hybrid algorithm is presented to address the closed path MDmTSP. The method is inspired by the method presented by Ángel A Juan et al. [7], where a Multi-Depot VRP is solved with a bias-randomized ILS. Similarly, the method presented consists of two stages. In the initial stage, numerous different solutions are generated using the biased-randomization technique. Subsequently, a few promising solutions are selected and further refined to produce near-optimal solutions. In Figure 1, the flow diagram of the proposed method can be observed.

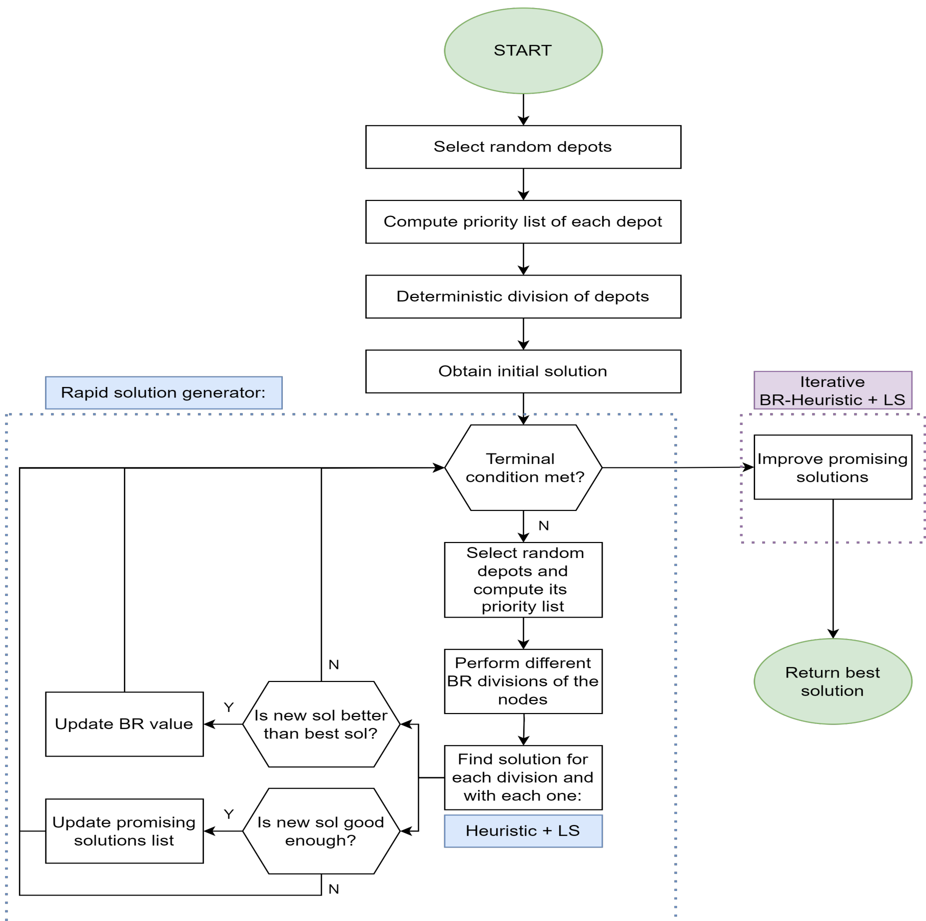


Fig. 1. Flow diagram of the proposed method.

The approach taken in the first stage of this problem consists of generating a large number of different solutions thanks to the implementation of bias randomization to the Nearest Neighbor heuristic (NN). To find the best routes to solve the problem, a new combination of nodes are selected as depots in every iteration, thereby increasing the number of possible solutions. Once the depots are selected, a priority list is computed to assign all nodes to the different depots. To compute the priority list of each depot, the marginal distance between a node and a depot is used. The marginal distance between a node and a depot is defined as the difference between the distance from the node to the depot and the distance of the node to the nearest depot among the rest. Mathematically, given a set of depots $\mathcal{D}$ and a set of nodes $\mathcal{N}$, the marginal distance between a node N^* and a depot D^* can be written as:

$$md(N^*, D^*) = d(N^*, D^*) - \min_{D \in \mathcal{D}} \{d(N^*, D)\}$$

where d represents a chosen distance.

For each depot, a node with a lower marginal distance has a higher priority of being selected once the division starts. After all marginal distances are computed, a list of nodes sorted by their marginal distance is created for each depot. In order to divide the nodes among the depots, the implemented method assigns a node to each depot in a bias-randomized manner. A probability is assigned to each element of the list, ensuring that the lower the marginal distance they have, the higher the probability of being selected. This probability assignment follows a geometric distribution, which is determined by a parameter β. This parameter is selected using a method that will be explained later. After a node is assigned to a depot, it is removed from all lists, and the method continues with the next depot. The depots rotate in order to distribute the nodes evenly. This approach generates compensated routes, as all depots will contain roughly the same number of nodes, ensuring a reasonable assignment. After this division, multiple variations are generated through a destruction-construction process, where a portion of the nodes assigned to each depot is removed and then reassigned in a bias-randomized manner, similar to how the original division was conducted. Thanks to the uncertainty of the division, this process allows for the computation of 10 different solutions for the same combination of depots to better evaluate if the depot selection can provide a good solution. In order to provide those solutions, once the second division is made, a combination of the NN algorithm and a fast shallow Local Search find a solution for the TSP composed of the depot and the nodes assigned to it. When a route is selected for the salesman departing from each depot, the total cost is computed as the sum of the costs of each route. Once the total cost is computed for each of the 10 solutions provided, the best solution among the ones generated is selected, in order to offer more variability to the second stage. Finally, two criteria are considered. The first one allows the method to be self-tuning. The parameter β previously mentioned is initially set to $0'5$. For each iteration, a new β is selected following a triangular distribution between 0 and 1, where the mode is β. This criterion consists of checking if the new solution generated is better than the previous best solution. If it is verified, the parameter β is set to the value that provided the new solution. The second criterion is used to save the promising solutions. A list consisting of a maximum number of "top" solutions is checked to see if the new solution is better than the worst saved promising solution. If it is verified, the worst saved solution is removed and the new solution is added to the list.

In the second stage, an enhancement procedure is employed to refine the saved promising solutions. In this method, a more thorough local search is applied to the solution, followed by the iterative application of a bias-randomized version of the NN algorithm to a new division of the nodes assigned to each depot following the same destruction-construction process previously explained. This enables the computation of various solutions while maintaining the same division of nodes as the promising solution. Additionally, this BR-NN algorithm is also self-tuning, as the parameter regarding bias randomization is updated as better solutions are found. Once the enhancement procedure is applied to all promising solutions, the best solution among them is selected.

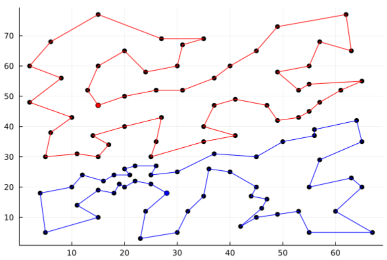
5 Results

The method has been implemented using Julia 1.10, while the computational experiments have been executed on a Windows 11 personal computer, with an i7-1165G7 CPU 2.80 GHz and 8 GB of RAM.

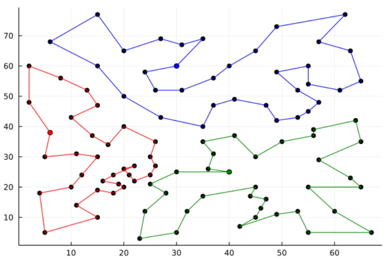
A key aspect of this algorithm is the balance of the solutions it generates. Once a solution is generated, all routes contain roughly the same number of nodes. Due to this reason, to the best of our knowledge, there is a lack of benchmark optimal solutions to test the method's efficiency. Therefore, a comparison of our results with optimal values would require the mathematical model and an exact method. M. Burger et al. [2] provided optimal solutions for several instances of the multi-depot mTSP without the constraint of balancing routes, which represents the closest approximation to a benchmark solution available. After comparing the results obtained with the method proposed in this paper and allowing around 25 seconds per instance, a 3% mean gap is obtained with the use of 2 depots, while adding the restriction of producing balanced routes, showcasing the effectiveness of this method. Nevertheless, as the results obtained by M. Burger et al. correspond to a less restrictive problem and considering that the increase in cost to the solution after adding this restriction is unknown, in this section, a comparison of the results obtained by the proposed method with a different number of depots is shown and analyzed. For further analysis of the results obtained, in Figure 2, a test solution expressed as a route map is displayed using 2 to 5 depots, and in Table 1, a comparison of the cost of each solution is shown. In this test, the problem is solved once per number of depots, and the amount of time given is higher as the number of depots increases, as the same time is given to improve the solution of every sub-route once the division is made. Nevertheless, an overall increase in the cost can be observed.

Table 1. Comparison between the costs of the solutions shown in Figure 2.

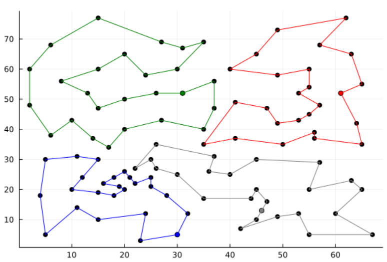
Instance	2 depots	3 depots	4 depots	5 depots
eil101	655'97	664'85	682'1	698'23



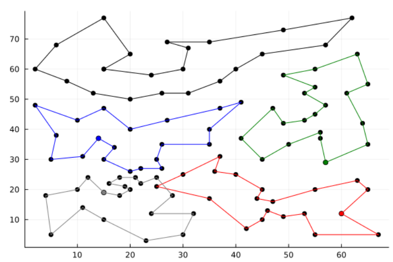
(a) Solution found with 2 depots.



(b) Solution found with 3 depots.



(c) Solution found with 4 depots.



(d) Solution found with 5 depots.

Fig. 2. Solutions found for instance eil101 with different number of depots.

In Table 2, a comparison between the costs of the obtained solutions is observed. In these tests, 25 seconds are permitted for every instance, regardless of the number of depots and every instance is solved with 10 different seeds. The cost of the solution with 2 depots is taken as the base, and the value for each instance with every number of depots represents the increased proportion between the obtained value and the value obtained with 2 depots. In this case, a value of 0'08 means that there is an increase of 8 % between the obtained result and the best result. When looking at the table, an increase in the overall cost of the solution can be observed. This increase is mainly due to two reasons.

The first reason is the need for greater time to generate a good division of the nodes as the number of depots increases. With each additional depot, the number of possible depot selections and the available number of different separations of nodes increases exponentially, thereby reducing the portion of possible solutions addressed and therefore reducing the quality of the solution.The second reason is the shape of the solutions generated. Due to the construction of the method, all routes generated in a solution are balanced according to the number of depots, which imposes a restriction that becomes more noticeable as the number of depots increases.

In Figure 3, the comparison between the solutions' costs can be observed in a more visual manner, where the overall increase of the solutions' cost can be noticed.

Table 2. Comparison between the result obtained for each instance with different number of depots.

Instance	**2 depots**	**3 depots**	**4 depots**	**5 depots**
berlin52	0	0'03	0'05	0'07
st70	0	0'01	0'04	0'10
eil101	0	0'01	0'03	0'05
kroB100	0	0'03	0'05	0'11
kroC100	0	-0'01	0'07	0'07
kroE100	0	0'02	0'03	0'08
kroA100	0	0'02	0'07	0'10
kroB200	0	0'01	0'06	0'09
a280	0	0'03	0'10	0'12
tsp225	0	0'05	0'04	0'09
ch150	0	0'02	0'05	0'10
Mean:	0	0'02	0'05	0'09

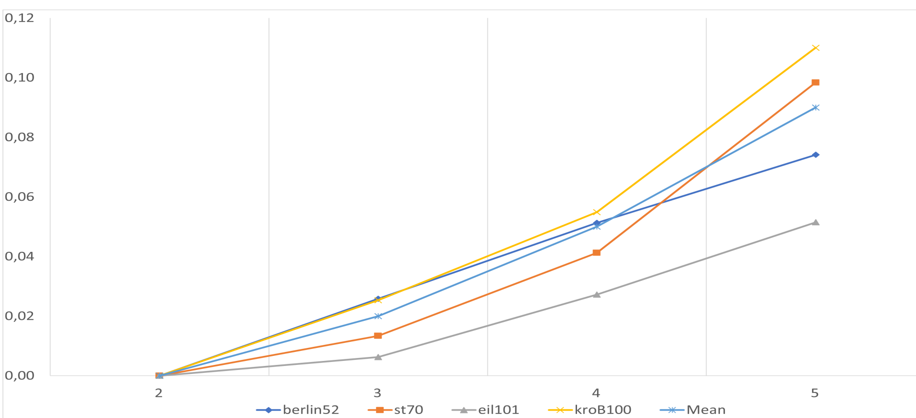


Fig. 3. Comparison between the result obtained for some instances according to the number of depots.

6 Managerial Insight

This method provides a simple way to finding solutions for plenty of real-life problems which can be modeled as a MDmTSP, which is of great interest in the managerial aspect. In this article, in addition to solving the problem, the best locations for the depots are selected. Therefore, this allows for the resolution of different real-life problems from those in which the depots are fixed. A realistic application for this method is in solving the facility location problem, where determining the optimal locations for new facilities while optimizing vehicle routing is necessary. Other uses include emergency response planning, where ambulance services, fire departments, and police agencies, need to strategically position their depots to minimize response times, or public transport planning, where new bus terminals or transit hubs need to strategically position these depots to improve service coverage. These are only some examples, as plenty of other uses can be found, including waste collection and recycling, food and beverage distribution or fleet management for companies with large fleets of vehicles, such as trucking companies, taxi services, and rental car companies, among many others.

Besides the variety of already mentioned applications, the simplicity of the method guarantees that minor changes can convert the algorithm for solving other very similar problems. For instance, if the depots are already known, a minor change would transform the method into a MDmTSP without depot location, which also holds numerous applications in real-life scenarios.

In addition, the algorithm's ability to provide high-quality solutions in a short period of time, enables for its implementation in problems where dynamism is present, such as what-if scenarios or instances where sudden changes would require new solutions in a short period of time.

7 Conclusion

This paper presents a simple and efficient approach to the closed-paths multi-depot mTSP, which generates balanced routes for multiple agents to visit all nodes in an instance. The method generates balanced solutions, meaning that all routes (if possible) have the same number of nodes. However, this factor leads to a worsening of the solutions' cost as the number of agents increases. Nevertheless, employing a higher number of agents results in each one traveling a shorter distance, which could be beneficial if additional constraints are imposed, such as those introduced by the use of electric vehicles (EVs), which have a maximum traveling distance determined by their battery capacity.

There is a vast amount of future work available regarding this subject. Numerous additional constraints can be integrated into this problem to enhance its realism, such as vehicle load capacity and node demand, thereby converting the problem into a VRP. Additionally, considerations related to the use of EVs and its associated constraints and consequences, as well as uncertainties or dynamism present in real-life scenarios, could be addressed.

Acknowledgments. This work has been partially supported by the Spanish Ministry of Science and Innovation (PID2022-138860NB-I00 and RED2022-134703-T).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Bajaj, A.S., Dhodiya, J.M.: Sustainable multiple travelling salesman problem solved by reference point aspiration level based multi objective quasi oppositional jaya algorithm under uncertain environment. Evolutionary Intelligence pp. 1–40 (2024)
2. Burger, M., Su, Z., De Schutter, B.: A node current-based 2-index formulation for the fixed-destination multi-depot travelling salesman problem. European Journal of Operational Research **265**(2), 463–477 (2018)
3. Cheikhrouhou, O., Khoufi, I.: A comprehensive survey on the multiple traveling salesman problem: Applications, approaches and taxonomy. Computer Science Review **40**, 100369 (2021)
4. He, P., Hao, J.K.: Memetic search for the minmax multiple traveling salesman problem with single and multiple depots. European Journal of Operational Research **307**(3), 1055–1070 (2023)
5. Hou, M., Liu, D.: A novel method for solving the multiple traveling salesmen problem with multiple depots. Chinese science bulletin **57**, 1886–1892 (2012)
6. Huang, J., Du, B., Zhang, Y., Quan, Q., Wang, B., Mu, L.: A pesticide spraying mission allocation and path planning with multicopters. IEEE Transactions on Aerospace and Electronic Systems (2024)
7. Juan, A.A., Pascual, I., Guimarans, D., Barrios, B.: Combining biased randomization with iterated local search for solving the multidepot vehicle routing problem. International Transactions in Operational Research **22**(4), 647–667 (2015)
8. Kara, I., Bektas, T.: Integer linear programming formulations of multiple salesman problems and its variations. European Journal of Operational Research **174**(3), 1449–1458 (2006)
9. Kumar, D.P., Rathinam, S., Darbha, S., Bihl, T.: Heuristic for min-max heterogeneous multi-vehicle multi-depot traveling salesman problem. In: AIAA SCITECH 2024 Forum. p. 0234 (2024)
10. Linganathan, S., Singamsetty, P.: Genetic algorithm to the bi-objective multiple travelling salesman problem. Alexandria Engineering Journal **90**, 98–111 (2024)
11. Oberlin, P., Rathinam, S., Darbha, S.: A transformation for a multiple depot, multiple traveling salesman problem. In: 2009 American Control Conference. pp. 2636–2641. IEEE (2009)
12. Pop, P.C., Cosma, O., Sabo, C., Sitar, C.P.: A comprehensive survey on the generalized traveling salesman problem. European Journal of Operational Research (2023)
13. Shao, Y., Xu, X.: Three-dimensional multi-uav trajectory design for cooperative video inspection and uploading. IEEE Transactions on Vehicular Technology (2023)