The multi-Depot multiple Set Orienteering Problem: An Integer Linear Programming Formulation

Ravi Kant and Abhishek Mishra
Department of CSIS, Birla Institute of Technology and Science, Pilani, 333031, India

Keywords: Traveling Salesman Problem (TSP), Orienteering Problem (OP), Set Orienteering Problem (SOP), Sub-tour Elimination Constraints (SECs).

Abstract: In this article, we introduce a novel variant of the single Depot multiple Set Orienteering Problem (sDmSOP), which we refer to as the multi-Depot multiple Set Orienteering Problem (mDmSOP). We suggest the integer linear program (ILP) of the mDmSOP also, and analyze the impact of the Sub-tour Elimination Constraints (SECs) based on the Miller–Tucker–Zemlin (MTZ) and the Gavish-Graves (GG) model on it. The mDmSOP is most frequently encountered in distribution logistics. In mDmSOP, a fleet of travelers is utilized to serve a set of customers from a number of depots, with each traveler associated with a specific depot. The challenge is to choose the routes for each traveler to maximize the profit within a specific budget, while the profit can be earned from a set of customers only once by visiting exactly one customer. We show the simulation results conducted on the General Algebraic Modeling System (GAMS) 39.0.2, which is used to model and analyze linear, non-linear, mixed-integer, and other complex optimization problems. The Generalized Traveling Salesman Problem (GTSP) instances of up to 200 vertices are taken as the input data set for the simulations. The results show that the MTZ-based formulation takes less time than the GG-based formulation to converge to the optimal solution for the mDmSOP.

1 INTRODUCTION

In recent years, the NP-hard routing problems with profits have received more attention due to their various real-world applications. These problems have mainly been classified as arc routing problems explored by (Archetti and Speranza, 2015) and node routing problems studied by (Archetti et al., 2014) and (Gunawan et al., 2016) recently. The Orienteering Problem (OP) suggested by (Golden et al., 1987) is one of the most explored profit-oriented variants of the Traveling Salesman Problem (TSP) in the literature and it belongs to the NP-hard complexity class. The Set Orienteering Problem (SOP) is a variant of the OP proposed by (Archetti et al., 2018). Recently the single Depot multiple Set Orienteering Problem (sDmSOP), a variant of the SOP has been proposed by (Kant et al., 2023), which has a single depot with multiple travelers associated with it and the objective is to find a path for each traveler to maximize the profit. These variants are mainly helpful for solving mass distribution supply-chain problems, and sensory network information retrieval.

Solving any NP-hard complexity class problem in an efficient time is one of the most desirable issues in recent years because the solution can be applied to all problems of that class to solve those problems efficiently, irrespective of the application domain of its researchers. So, choosing the correct Sub-tour Elimination Constraints (SECs) is a problem encountered in all NP-hard routing problems. From the Vehicle Routing Problem (VRP) to the OP, a lot of research is done on the relative comparisons of the formulations based on the SECs proposed by (Dantzig et al., 1954), (Miller et al., 1960) (MTZ) and (Gavish and Graves, 1978) (GG). (Lalla-Ruiz et al., 2016) show that the improved MTZ SECs work better than the SECs given by (Desrochers and Laporte, 1991) and (Kara et al., 2004) for the multi-depot open vehicle routing problem. (Vansteenwegen et al., 2011) proposed an integer programming (IP) formulation for the Orienteering Problem, which used the MTZ-based SECs, while the Set Orienteering Problem proposed by (Archetti et al., 2018) implemented SECs using the GG-based formulation. Here, the SECs chosen for the formulation are based on the comparative analysis done by (Öncan et al., 2009); the study shows that the GG-based SECs yield better results than the MTZ-

Kant, R. and Mishra, A.
The multi-Depot multiple Set Orienteering Problem: An Integer Linear Programming Formulation.
DOI: 10.5220/0012420500003639

In *Proceedings of the 13th International Conference on Operations Research and Enterprise Systems (ICORES 2024)*, pages 350-355
ISBN: 978-989-758-681-1; ISSN: 2184-4372

based SECs for 24 Asymmetric Traveling Salesman Problem formulations. (Bazrafshan et al., 2021) concluded in their comparative study of sub-tour elimination methods that it is not specified which SECs are better suited for a problem until we check the performance of different SECs. Hence, a comparative study must be performed to choose the correct SECs for the specific formulation.

In this paper, we formally define and propose the integer linear program (ILP) for the multi-Depot multiple Set Orienteering Problem (mDmSOP) and compare the computational time of the mDmSOP formulation using sub-tour elimination constraints (SECs) proposed by (Miller et al., 1960) and (Gavish and Graves, 1978). As we already know, the SECs are the most computationally expensive part of the ILP formulation. The reason for choosing these SECs is the same number of variables and constraints of the order $O(n^2)$ for TSP, but here, we simulate both SECs for the mDmSOP formulation and identify which one works best for our problem. The mDmSOP is important to study as many supply-chain problems can not be modeled using a single depot and multiple travelers like the sDmSOP. In the mDmSOP, we are given M' depots, having one traveler each. The vertices are divided into mutually exclusive clusters, and there is an associated profit with each cluster. The traveler has to reach exactly one vertex in the cluster to earn the profit of that cluster. There is a budget constraint B, that bounds the maximum distance traversed by all the travelers.

The following is the summary of our contribution to this paper; we propose a mathematical formulation in section 2. The comparative results of the mDmSOP formulation based on the MTZ-based SECs (ILP 1) and the GG-based SECs (ILP 2) are shown in section 3, and the conclusion and future scope of research is given in section 4.

2 PROBLEM DEFINITION AND FORMULATION

In this section, a formal definition and mathematical formulation of the mDmSOP are presented as follows:

We have an undirected complete weighted graph $G(V,E)$, where the vertex set and edge set is represented by $V = \{v_1, v_2, \cdots, v_n\} \cup \{v_{n+1}, \cdots, v_{n+m}\}$ and $E = \{e_{pq} : \forall (p,q) \in V\}$, respectively. The last m vertices in the vertex set represent the depots from where the m travelers start. Let the index set for travelers be represented by $M = \{1, 2, \cdots, m\}$ and the last m vertices by the set M', while C_{pq} represents the cost to travel from vertex p to q of edge e_{pq}. Its cost matrix satisfies the triangle inequality, i.e., $C_{pr} \leq C_{pq} + C_{qr}\ \forall (p,q,r) \in V$. For vertices p and q, C_{pq} is chosen as the minimum cost of all possible paths between vertices p and q; from shortest path problems, the triangle inequality is guaranteed by minimum cost criterion. For each traveler, the cost incurred is the sum of the costs of all edges traversed by that traveler. Combining the sum of costs for all m salesmen, we get the overall cost. An upper bound of B is placed on the total cost that is permitted in the final solution of our problem. Every traveler is required to reach its starting position at the end of the tour, and every traveler must visit at least one vertex other than the depot. The vertex set V is partitioned into clusters represented by a set $S = \{S_i : i = 1, \cdots, r\}$, such that $S_i \cap S_j = \emptyset, \forall i \neq j$ and $\bigcup_{i=1}^{i=r} S_i = V$, where r represents the total number of clusters. We also have $S_\mu \subseteq S$ where the set S_μ represents the last m clusters in S. For each $S_i \in S$, we have an associated profit P_i that is obtained once a traveler visits any vertex in S_i. The cost of any cluster in S_μ is defined to be 0 as they are the starting point for their traveler. For each cluster $S_i \in S$, we have the restriction that only one traveler visits that cluster and only one vertex in that cluster, visiting two or more vertices in the same set does not give any additional profit but increases our overall cost. However, this restriction is not there for the last m clusters, as travelers are required to reach their starting depot after the completion of their tour. We define the following decision variables for our formulation:

1. x_{jpq}: Binary decision variable which takes value 1 iff edge (p,q) is visited by traveler j, otherwise 0.
2. y_{jp}: Binary decision variable which takes value 1 iff vertex p is visited by traveler j, otherwise 0.
3. z_{ji}: Binary decision variable which takes value 1 iff cluster i is visited by traveler j, otherwise 0.

The goal of the problem is to maximize the profit that can be achieved by all travelers without violating the constraints of the problem.

The mathematical formulation of the mDmSOP can be formalized as follows:

$$maximize \sum_{j \in M} \sum_{i \in S} P_i z_{ji}, \tag{1}$$

subject to:

$$x_{jpq}, y_{jp}, z_{ji} \in \{0,1\}, \quad \forall j \in M, \forall (p,q) \in V, \tag{2}$$

$$\sum_{j \in M} \sum_{p \in V} \sum_{q \in V} x_{jpq} C_{pq} \leq B, \tag{3}$$

$$y_{j(j+n)} = 1, \quad \forall j \in M, \tag{4}$$

$$\sum_{p \in V-\{q\}} x_{jpq} = y_{jq}, \quad \forall j \in M, \forall q \in V, \tag{5}$$

$$\sum_{p \in V-\{q\}} x_{jqp} = y_{jq}, \quad \forall j \in M, \forall q \in V, \tag{6}$$

$$\sum_{p \in S_i} y_{jp} = z_{ji}, \quad \forall j \in M, \forall i \in S, \tag{7}$$

$$\sum_{j \in M} z_{ji} \leq 1, \quad \forall i \in S, \tag{8}$$

$$0 \leq u_{pq} \leq (n-m+1) \sum_{j=1}^{m} x_{jpq}, \quad \forall (p,q) \in (V-M')^2, \tag{9}$$

$$\sum_{q \in V, p \neq q} u_{pq} - \sum_{q \in V-M, p \neq q} u_{qp} = \sum_{j=1}^{m} y_{jp}, \quad \forall p \in V-M', \tag{10}$$

$$1 \leq u_{jp} \leq n, \forall j \in M, \quad \forall p \in V, \tag{11}$$

$$\begin{aligned} u_{jp} - u_{jq} + 1 \leq n(1-x_{jpq}), \quad \forall j \in M, \\ \forall (p,q) \in \{V-M' | p \neq q\}. \end{aligned} \tag{12}$$

The objective function (1) maximizes collected profit from the clusters visited, equation (2) defines the nature of the variables, equation (3) ensures that the overall travel cost should not exceed the budget B. Equation (4) ensures that the traveler associated with a depot should finish their journey at the same depot. Equations (5) and (6) ensure that the in-degree is equal to the out-degree of a vertex. Equation (7) implies that for any cluster i and traveler j, j visits the cluster i at most once, while equation (8) ensures that no cluster can be visited more than once by any traveler. Equations (9) and (10) are based on (Gavish and Graves, 1978). They used the arcs of network flow to get the SECs for the mDmSOP while (11) and (12) are based on (Miller et al., 1960), where node potential u_{jp} specifies the order in which the vertices are visited by traveler j, $u_{jp} \leq u_{jq}$ implies that traveler j visits vertex p before vertex q.

3 COMPARATIVE RESULTS

In this section, we present the computational results of the tests we made to evaluate the performance of the mDmSOP. The simulation is done on Intel® Xeon(R) Silver 4316 CPU @ 2.30GHz × 80 with 256 GB of RAM using all 40 threads. The mathematical formulations were solved using the GAMS 39.0.2.

In section 3.1, we describe how the instances are generated for the mDmSOP, and the computational results are shown in section 3.2.

3.1 Test Instances

To analyze the comparative results of the above formulations, the Generalized Traveling Salesman Problem (GTSP) instances suggested by (Noon, 1988) are used. We modified the GTSP instances for our formulation as follows:

1. Move the depot vertices from non-depot clusters to depot clusters.
2. Sort the non-depot clusters in ascending order of the number of vertices in the clusters.
3. Iterate over the list, and if there is an empty cluster, find the first vertex from a non-empty cluster with a size greater than one and put it into the empty cluster found.

This algorithm generates clusters that satisfy the constraints of our problem.

The profit is assigned to the clusters using g_1 and g_2 schemes used by (Pěnička et al., 2019). In the g_1 scheme, the profit of a cluster is equal to the number of vertices in that cluster. Whereas in the g_2 scheme for each vertex numbered k, the profit assigned is $(1 + 7141 \times k)$mod 100, and the profit assigned to a cluster is the sum of profits of the vertices in the cluster. In each case, the clusters containing the depot are assigned a profit of 0.

3.2 Computational Results

The results of the simulation are shown in Table 1 and Table 2 with the following criteria:

1. Table 1: threads = 0.
2. Table 2: threads = 0 - Set 1 has a 5% relative gap, and Set 2 has a 20% relative gap.

Here threads = 0 means the system is using all the available threads. All the instances of Table 1 are solved till we find the optimal solution, but for Table 2, the GAMS stops if the solution is found in a relative gap of 5% for Set 1 and 20% for Set 2, respectively. Set 1 contains the instances which can be

Table 1: ILP comparison with optimal solutions on small instances with $w < 1$.

	Instance	**n**	**m**	**Pg**	**w**	**Opt.**	**ILP 1**			**ILP 2**		
							Sol.	Time	Gap	Sol.	Time	Gap
	11berlin52	52	2	g_1	0.4	27	27	**8.69**	0.00	27	7.57	0.00
	11berlin52	52	2	g_1	0.6	34	34	**1392.51**	0.00	34	325.71	0.00
	11berlin52	52	2	g_1	0.8	45	45	**15457.76**	0.00	45	2644.50	0.00
	11berlin52	52	2	g_2	0.4	1276	1276	6.99	0.00	1276	8.48	0.00
	11berlin52	52	2	g_2	0.6	1571	1571	**2360.78**	0.00	1571	384.84	0.00
	11berlin52	52	3	g_1	0.4	36	36	**6.39**	0.00	36	5.78	0.00
	11berlin52	52	3	g_1	0.6	43	43	42.67	0.00	43	114.42	0.00
	11berlin52	52	3	g_1	0.8	47	47	**36.34**	0.00	47	20.79	0.00
	11berlin52	52	3	g_2	0.4	1788	1788	**8.29**	0.00	1788	7.47	0.00
	11berlin52	52	3	g_2	0.6	2083	2083	**388.39**	0.00	2083	142.51	0.00
	11berlin52	52	3	g_2	0.8	2265	2265	54.48	0.00	2265	445.27	0.00
	11eil51	51	2	g_1	0.4	29	29	5.92	0.00	29	8.44	0.00
	11eil51	51	2	g_1	0.6	38	38	69.97	0.00	38	239.20	0.00
	11eil51	51	2	g_1	0.8	45	45	76.09	0.00	45	136.23	0.00
	11eil51	51	2	g_2	0.4	1552	1552	4.78	0.00	1552	5.69	0.00
	11eil51	51	2	g_2	0.6	1931	1931	121.88	0.00	1931	376.14	0.00
	11eil51	51	2	g_2	0.8	2226	2226	179.40	0.00	2226	350.85	0.00
	11eil51	51	3	g_1	0.4	27	27	6.15	0.00	27	12.53	0.00
	11eil51	51	3	g_1	0.6	37	37	39.53	0.00	37	80.06	0.00
	11eil51	51	3	g_1	0.8	42	42	**418.33**	0.00	42	310.25	0.00
	11eil51	51	3	g_2	0.4	1483	1483	**6.43**	0.00	1483	5.56	0.00
	11eil51	51	3	g_2	0.6	1862	1862	54.49	0.00	1862	158.43	0.00
	11eil51	51	3	g_2	0.8	2077	2077	282.39	0.00	2077	839.42	0.00
	16eil76	76	2	g_1	0.4	39	39	591.15	0.00	39	1575.45	0.00
	16eil76	76	2	g_1	0.6	54	54	9222.96	0.00	54	14121.98	0.00
	16eil76	76	2	g_2	0.4	1939	1939	433.91	0.00	1939	1487.15	0.00
	16eil76	76	2	g_2	0.6	2621	2621	13025.47	0.00	2621	33608.96	0.00
	16eil76	76	3	g_1	0.4	43	43	49.72	0.00	43	368.46	0.00
	16eil76	76	3	g_1	0.6	57	57	1949.76	0.00	57	8194.79	0.00
	16eil76	76	3	g_1	0.8	65	65	13809.58	0.00	65	32938.12	0.00
	16eil76	76	3	g_2	0.4	2042	2042	50.74	0.00	2042	366.64	0.00
	16eil76	76	3	g_2	0.6	2696	2696	3645.93	0.00	2696	21970.50	0.00
	16eil76	76	3	g_2	0.8	3070	3070	25086.16	0.00	3070	151192.61	0.00
Avg.								**2693.76**			**8256.21**	

solved using GAMS optimally, and Set 2 contains the instances which have less than 200 vertices and which can not be solved using GAMS optimally. The results are presented in Table 1 and Table 2. The organization of Table 1 is as follows: The first five columns represent the GTSP instance name, the number of vertices (n) in the GTSP instance, the number of travelers (m) used, the rule to generate the profit (P_g), and the value of w. Budget is calculated as $B = w \times T_{\max}$, where w is a multiplicative factor so that we can adjust the budget according to our need and $T_{\max}$ is the solution of the GTSP instance, Opt. column shows the optimal solution for the mDmSOP. The last six columns represent the solution, time, and relative gap for the mDmSOP formulation based on the MTZ and the GG SECs respectively.

4 CONCLUSION

In this paper, we introduce a generalization of the single Depot multiple Set Orienteering Problem (sDmSOP) which has multiple depots and each depot has one traveler associated with it; the goal of this problem is to maximize profit from mutually exclusive clusters using multiple travelers with the respective starting and ending depot with a cumulative budget shared by all the travelers. Each cluster is associated with a fixed profit that is determined by two rules termed g_1 and g_2, and the profit can only be earned if a traveler visits exactly one vertex from a cluster. The proposed variant has applications mainly in the supply chain, where a distributor has more than one service point from where the distributor can supply the

Table 2: ILP comparison on small and medium instances with w = 1.

	Instance	**n**	**m**	**Pg**	**Opt.**	**ILP 1**			**ILP 2**		
						Sol.	Time	Gap	Sol.	Time	Gap
Set 1	11berlin52	52	2	g_1	50	48	1.81	4.00	48	16.22	4.00
	11eil51	51	2	g_1	49	47	4.84	4.08	47	5.91	4.08
	14st70	70	2	g_1	68	67	9.93	1.47	67	34.55	1.47
	16eil76	76	2	g_1	74	71	125.56	4.05	71	244.69	4.05
	11berlin52	52	2	g_2	2375	2264	**5.36**	4.67	2264	2.52	4.67
	11eil51	51	2	g_2	2365	2293	6.64	3.04	2293	7.96	3.04
	14st70	70	2	g_2	3266	3182	12.81	2.57	3182	16.50	2.57
Avg.							**23.85**			**46.91**	
Set 2	20rat99	99	2	g_1	97	85	102.28	12.37	83	281.75	14.43
	20rd100	100	2	g_1	98	84	9.91	14.29	89	18.45	9.18
	21eil101	101	2	g_1	99	85	10.00	14.14	87	63.88	12.12
	21lin105	105	2	g_1	103	89	11.98	13.59	92	45.68	10.68
	25pr124	124	2	g_1	122	103	289.63	15.57	102	543.62	16.39
	26bier127	127	2	g_1	125	106	**31.69**	15.20	110	21.64	12.00
	26ch130	130	2	g_1	128	109	37.36	14.84	119	410.47	7.03
	28pr136	136	2	g_1	134	114	64.90	14.93	119	482.91	11.19
	29pr144	144	2	g_1	142	123	**382.59**	13.38	126	285.10	11.27
	30ch150	150	2	g_1	148	130	376.93	12.16	126	946.24	14.86
	30kroA150	150	2	g_1	148	127	505.32	14.19	124	1360.52	16.22
	30kroB150	150	2	g_1	148	130	**343.76**	12.16	137	122.13	7.43
	31pr152	152	2	g_1	150	128	600.80	14.67	126	1004.65	16.00
	32u159	159	2	g_1	157	131	493.96	16.56	131	849.72	16.56
	39rat195	195	2	g_1	193	193	**49.44**	0.00	192	23.86	0.52
	40d198	198	2	g_1	196	183	**42.91**	6.63	191	35.96	2.55
	20rat99	99	2	g_2	4793	4228	114.81	11.79	4073	488.31	15.02
	20rd100	100	2	g_2	4871	4304	8.58	11.64	4151	12.79	14.78
	21eil101	101	2	g_2	4890	4200	10.99	14.11	4293	36.15	12.21
	21lin105	105	2	g_2	5076	4287	10.48	15.54	4294	15.35	15.41
	25pr124	124	2	g_2	6043	5172	230.56	14.41	5230	1106.06	13.45
	26bier127	127	2	g_2	6175	5888	15.67	4.65	5458	21.16	11.61
	26ch130	130	2	g_2	6276	5256	43.64	16.25	5318	213.55	15.26
	28pr136	136	2	g_2	6585	5692	386.29	13.56	5490	628.93	16.63
	29pr144	144	2	g_2	6993	6021	**322.39**	13.90	5951	222.97	14.90
	30ch150	150	2	g_2	7246	6052	**494.60**	16.48	6334	85.90	12.59
	30kroA150	150	2	g_2	7246	6089	708.48	15.97	6070	1322.84	16.23
	30kroB150	150	2	g_2	7246	6459	34.70	10.86	6455	826.44	10.92
	31pr152	152	2	g_2	7325	6237	**646.70**	14.85	6181	346.05	15.62
	32u159	159	2	g_2	7743	6519	446.09	15.81	6561	517.54	15.27
	39rat195	195	2	g_2	9541	9175	**23.62**	3.84	9519	23.20	0.23
	40d198	198	2	g_2	9706	8330	17.37	14.18	9476	35.15	2.37
Avg.							**214.64**			**387.47**	

products to customers and gain the maximum profit while providing the customer with a better price for the product.

Out of 33 instances of Table 1, only 10 instances take less time to find the optimal solution for the mDmSOP formulation using GG-based SECs; the average time taken by GG-based SECs formulation is 3.06 times more than the average time taken by the MTZ-based SECs formulation for the instances of Table 1. Only in 1 instance of Set 1 of Table 2, the GG-based SECs formulation is able to find a solution within a 5% relative gap in a lesser time than the MTZ-based SECs formulation. For Set 2 of Table 2, the GG-based SECs formulation finds a solution in less time than the MTZ-based SECs formulation in only 9 instances out of 32 instances. Moreover, the average time taken in the MTZ-based SECs formulation is lesser than the GG-based SECs formulation of the mDmSOP, so we can conclude that the MTZ-based SECs work better in most of the instances for the mDmSOP. Future research directions may include comparing a new variant of the mDmSOP with an individual traveler budget, rather than a cumulative budget shared by all the travelers, and developing an efficient algorithm dedicated to solving both the individual and cumulative budget variation of the mDmSOP.

REFERENCES

Archetti, C., Carrabs, F., and Cerulli, R. (2018). The set orienteering problem. *European Journal of Operational Research*, 267(1):264–272.

Archetti, C. and Speranza, M. G. (2015). Chapter 12: Arc routing problems with profits. In *Arc routing: problems, methods, and applications*, pages 281–299. SIAM.

Archetti, C., Speranza, M. G., and Vigo, D. (2014). Chapter 10: Vehicle routing problems with profits. In *Vehicle routing: Problems, methods, and applications, second edition*, pages 273–297. SIAM.

Bazrafshan, R., Hashemkhani Zolfani, S., and Mirzapour Al-e hashem, S. M. J. (2021). Comparison of the subtour elimination methods for the asymmetric traveling salesman problem applying the seca method. *Axioms*, 10(1):19.

Dantzig, G., Fulkerson, R., and Johnson, S. (1954). Solution of a large-scale traveling-salesman problem. *Journal of the operations research society of America*, 2(4):393–410.

Desrochers, M. and Laporte, G. (1991). Improvements and extensions to the miller-tucker-zemlin subtour elimination constraints. *Operations Research Letters*, 10(1):27–36.

Gavish, B. and Graves, S. C. (1978). The travelling salesman problem and related problems.

Golden, B. L., Levy, L., and Vohra, R. (1987). The orienteering problem. *Naval Research Logistics (NRL)*, 34(3):307–318.

Gunawan, A., Lau, H. C., and Vansteenwegen, P. (2016). Orienteering problem: A survey of recent variants, solution approaches and applications. *European Journal of Operational Research*, 255(2):315–332.

Kant, R., Mishra, A., and Sharma, S. (2023). The single depot multiple set orienteering problem.

Kara, I., Laporte, G., and Bektas, T. (2004). A note on the lifted miller–tucker–zemlin subtour elimination constraints for the capacitated vehicle routing problem. *European Journal of Operational Research*, 158(3):793–795.

Lalla-Ruiz, E., Expósito-Izquierdo, C., Taheripour, S., and Voß, S. (2016). An improved formulation for the multi-depot open vehicle routing problem. *OR spectrum*, 38(1):175–187.

Miller, C. E., Tucker, A. W., and Zemlin, R. A. (1960). Integer programming formulation of traveling salesman problems. *Journal of the ACM (JACM)*, 7(4):326–329.

Noon, C. E. (1988). *The generalized traveling salesman problem*. PhD thesis, University of Michigan.

Öncan, T., Altınel, I. K., and Laporte, G. (2009). A comparative analysis of several asymmetric traveling salesman problem formulations. *Computers & Operations Research*, 36(3):637–654.

Pěnička, R., Faigl, J., and Saska, M. (2019). Variable neighborhood search for the set orienteering problem and its application to other orienteering problem variants. *European Journal of Operational Research*, 276(3):816–825.

Vansteenwegen, P., Souffriau, W., and Van Oudheusden, D. (2011). The orienteering problem: A survey. *European Journal of Operational Research*, 209(1):1–10.