mathematics

Article

Compact Integer Programs for Depot-Free Multiple Traveling Salesperson Problems

José Alejandro Cornejo-Acosta [1,2], Jesús García-Díaz [1,3,*], Julio César Pérez-Sansalvador [1,3] and Carlos Segura [4]

[1] Coordinación de Ciencias Computacionales, Instituto Nacional de Astrofísica, Óptica y Electrónica, Puebla 72840, Mexico ; alexcornejo@inaoep.mx (J.A.C.-A.); jcp.sansalvador@inaoep.mx (J.C.P.-S.)
[2] División de Ingeniería Informática, Tecnológico Nacional de México/ITS de Purísima del Rincón, Guanajuato 36425, Mexico
[3] Consejo Nacional de Humanidades, Ciencias y Tecnologías, Ciudad de México 03940, Mexico
[4] Área de Computación, Centro de Investigación en Matemáticas, Guanajuato 36240, Mexico; carlos.segura@cimat.mx
* Correspondence: jesus.garcia@conahcyt.mx

Abstract: Multiple traveling salesperson problems (mTSP) are a collection of problems that generalize the classical traveling salesperson problem (TSP). In a nutshell, an mTSP variant seeks a minimum cost collection of m paths that visit all vertices of a given weighted complete graph. This paper introduces novel compact integer programs for the depot-free mTSP (DFmTSP). This fundamental variant models real scenarios where depots are unknown or unnecessary. The proposed integer programs are adapted to the main variants of the DFmTSP, such as closed paths, open paths, bounding constraints (also known as load balance), and the minsum and minmax objective functions. Some of these integer programs have $\mathcal{O}(n^2m)$ binary variables and $\mathcal{O}(n^2)$ constraints, where m is the number of salespersons and $n = |V(G)|$. Furthermore, we introduce more compact integer programs with $\mathcal{O}(n^2)$ binary variables and $\mathcal{O}(n^2)$ constraints for the same problem and most of its main variants. Without losing their compactness, all the proposed programs are adapted to fixed-destination multiple-depots mTSP (FD-MmTSP) and a combination of FD-MmTSP and DFmTSP, where fewer than m depots are part of the input, but the solution still consists of m paths. We used off-the-shelf optimization software to empirically test the proposed integer programs over a classical benchmark dataset; these tests show that the proposed programs meet desirable theoretical properties and have practical advantages over the state of the art.

Keywords: integer programming; multiple traveling salesperson problem; depot-free mTSP

MSC: 90C10; 90-10; 90B06

Citation: Cornejo-Acosta, J.A.; García-Díaz, J.; Pérez-Sansalvador, J.C.; Segura, C. Compact Integer Programs for Depot-Free Multiple Traveling Salesperson Problems. *Mathematics* **2023**, *11*, 3014. https://doi.org/10.3390/math11133014

Academic Editors: Shiv Raj Singh, Dharmendra Yadav and Himani Dem

Received: 21 June 2023
Revised: 4 July 2023
Accepted: 5 July 2023
Published: 6 July 2023

1. Introduction

The multiple traveling salesperson problem has received different labels over time, mainly because it is not a single problem but a collection of them. Thus, this paper refers to this collection as the multiple traveling salesperson problems (mTSP). These problems are relevant in several social situations, including cooperative missions, transportation, delivery, disaster management, precision agriculture, and many others [1].

Each variant of mTSP is a generalization of the $\mathcal{NP}$-hard traveling salesperson problem (TSP). The goal of TSP is to find a minimum-cost closed path a salesperson must follow to visit a set of given cities. In more detail, given a weighted complete graph $G = (V, E)$, the TSP seeks a closed path that visits all vertices once, minimizing the path's cost [2–4]. In the mTSP, the input is a weighted complete graph $G = (V, E)$ and a positive integer m; its goal is to find a set of m paths such that all vertices are visited once by some salesperson [4,5]. If the input graph is undirected (resp., directed), the problem is symmetric (resp., asymmetric).

An mTSP variant receives particular adjectives depending on its characteristics: depot free (DF), single depot (S), multiple depots (M), closed paths (CP), open paths (OP), bounding constraints, etc. In most cases, the objective function to minimize is the sum of the paths' costs (minsum) or the largest path (minmax or makespan); other objective functions might be considered, such as the cost of the largest edge (bottleneck).

The most widely studied members of the mTSP collection are single-depot mTSP (SmTSP) and multiple-depots mTSP (MmTSP). However, the depot-free mTSP (DFmTSP) variant has received less attention. In SmTSP, all salespersons must start and finish their path at a specific vertex (the depot), which is part of the input. In the fixed-destination multiple-depots mTSP (FD-MmTSP), m depots are part of the input, and each salesperson must start and finish their path at their respective depot. In the non-fixed-destination multiple-depots mTSP (NFD-MmTSP), each salesperson can finish their path at a different depot. In DFmTSP, the variant studied in this paper, the depot concept is not involved. Therefore, it seeks a disjoint collection of closed paths that visit all vertices. In all these variants, every solution consists of exactly m paths, and the path followed by each salesperson is closed. Nevertheless, if the salespersons are constrained to follow open paths (i.e., they do not need to return to their depot), we refer to the problem as an open-paths (OP) variant. To clarify the difference between these variants, Figure 1 shows a set of optimal solutions for DFmTSP, SmTSP, and FD-MmTSP. In all these examples, each path must have between three and five vertices; we refer to these as bounding constraints.

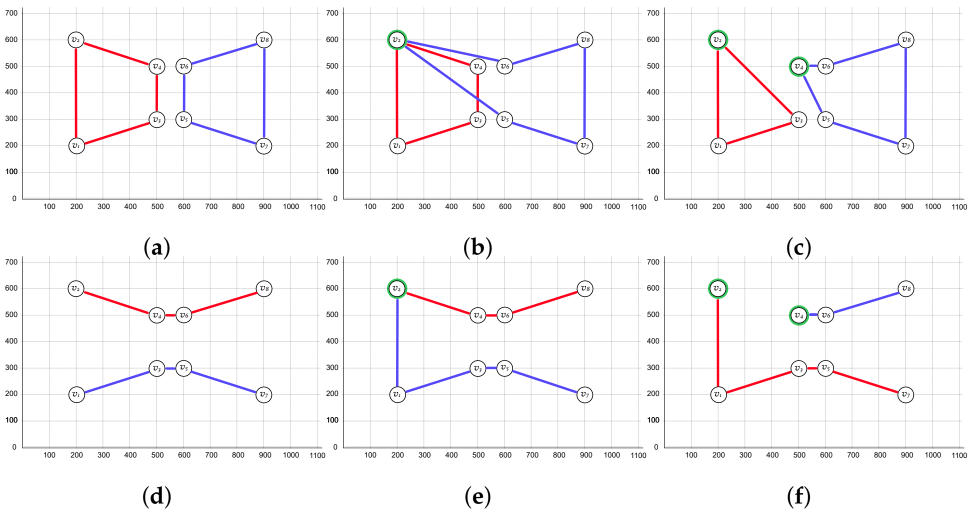


Figure 1. Optimal solutions for (**a**) closed-paths depot-free mTSP (CP-DFmTSP), (**b**) closed-paths single-depot mTSP (CP-SmTSP), and (**c**) closed-paths fixed-destination multiple-depots mTSP (CP-FD-MmTSP). The objective function is minsum, the number of salespersons is two ($m = 2$), each path must have between three and five vertices (bounding constraints), the cost of each edge equals the euclidean distance between its vertices, and the depots are marked in green. Subfigures (**d**–**f**) correspond to the respective open-paths (OP) variants.

Many variants of mTSP have been studied over the last decades. However, most of the attention has been paid to SmTSP and MmTSP. As we argue in the next section, DFmTSP variants should receive more attention, and mathematical modeling is an important first step. Therefore, this paper introduces novel compact integer programs (IPs) for DFmTSP and its main variants: CP, OP, bounding constraints, and minsum and minmax objective functions. To our knowledge, these mathematical models are the first reported with such a wide scope. One of the main features of the proposed mathematical formulations is the presence of dummy depots, special vertices that allow us to relate DFmTSP to FD-MmTSP.

The remaining sections of the document are organized as follows. Section 2 presents a literature review of the mTSP collection. Section 3.1 introduces a set of IPs for DFmTSP and its main variants: CP, OP, bounding constraints, minsum and minmax objective functions. All IPs from Section 3.1 extend a state-of-the-art IP for FD-MmTSP. Section 3.2 introduces more compact IPs for DFmTSP and most of its main variants: CP, OP, bounding constraints, and minsum objective function. At the end of Sections 3.1 and 3.2, we show that a slight modification of the proposed IPs leads to valid models for FD-MmTSP, and a combination between this problem and DFmTSP. Section 4 shows empirical tests and comparisons among the proposed IPs and a state-of-the-art formulation. Finally, Sections 5 and 6 present a discussion and concluding remarks.

2. Related Work

Although there are some surveys on mTSP [1,6,7], they are not structured in chronological order; order that might shed some light on how DFmTSP has received less attention than other variants. Therefore, this section shows a general overview of how the study of mTSP has evolved. To maintain simplicity, we stuck to the broad categories: SmTSP, MmTSP, and DFmTSP. Namely, we omitted sophisticated requirements studied in some examined papers (objective functions, paths' properties, time windows, etc.). Although mTSP is a generalization of TSP [2,3,5] and a particular case of vehicle routing problem (VRP) [8,9], in this paper, we mainly considered papers directly related to mTSP. Additionally, we only included the most common single-objective variants of the problem. Nevertheless, to talk about mTSP, we must begin by talking about TSP.

2.1. TSP and SmTSP

TSP is among the most popular $\mathcal{NP}$-hard combinatorial optimization problems. Its first explicit appearance in the scientific literature dates from 1954 [2], and its first mathematical formulations from 1959 and 1960 [3,4]. Interestingly, the authors of these early papers stated the problem using the depot concept, a special vertex where the salesperson starts and finishes their path. Nevertheless, it is easy to observe that the depot plays only a symbolic and didactic role, i.e., it is equivalent to stating that the path must be closed. However, the first member of the mTSP collection to be studied was SmTSP; in this version of the problem, all salespersons start and finish their path at the same vertex (the depot), which is part of the input. The most usual objective function to minimize in both TSP and mTSP is minsum, i.e., the total length of the paths. Naturally, mTSP with one salesperson ($m = 1$) is equivalent to TSP; therefore, mTSP is $\mathcal{NP}$-hard too.

2.2. mTSP from 1960 to 1975

Although the first IP for SmTSP dates from 1960 [4], the problem started gaining more attention between 1973 and 1975, when more efficient mathematical formulations were introduced [5,10]; the authors did not refer to the problem as SmTSP but as mTSP. For that reason, many use both names interchangeably. However, to avoid confusion, we refer to this variant only as SmTSP. Something remarkable about SmTSP is that it can be transformed into TSP by adding some extra vertices to the original graph [10].

2.3. mTSP from 1976 to 1995

SmTSP and many of its variants continued being modeled using integer programming. Most of these formulations were based on transformations to TSP [11–19], and only a few were direct [20,21]. Some variants included the case where at most m salespersons are in the solution; some refer to such variant as the SmTSP with fixed charges [13], and others as variable SmTSP [22]. In this variant, each salesperson incurs some cost that is considered in the objective function. Notice that the variant identified as SmTSP is where exactly m salespersons are in the solution. The exact algorithms designed for this problem within this period were based mainly on Benders decomposition [14], cutting planes [20], and branch and bound [23]. In this same period, some heuristics for SmTSP were introduced too [19,24–27]. By 1995, SmTSP with a minsum objective function was the most studied member of the mTSP collection; only the MmTSP with two salespersons ($m = 2$) was mentioned and reduced to TSP in 1980 [15]. It was until 1995 that MmTSP was reduced to TSP [18]. In this period, only a tabu search metaheuristic approach was developed for SmTSP [28].

2.4. *mTSP from 1996 to 2005*

MmTSP started gaining more attention. More heuristics and metaheuristics considering minsum and minmax objective functions for SmTSP, MmTSP, and DFmTSP were introduced too, for instance, neural networks [29–31], genetic algorithms [32–34], particle swarm optimization [33], evolutionary strategies [33], and simulated annealing [35]. It is crucial to emphasize that during this period, DFmTSP started being spotted by some authors [32,33,36]. Although some of them referred to the problem as mTSP or SmTSP, a closer inspection reveals that the problem they worked with was actually DFmTSP.

2.5. *mTSP from 2006 to Date*

The interest in SmTSP, MmTSP, DFmTSP, and their variants continued growing. The interest in the minmax objective function grew too. However, most of the efforts were still hoarded by SmTSP and the minsum objective function. In this period, many heuristics [37–41], exact algorithms [42–47], and IPs [6,42,44,48–54] were proposed. Metaheuristics dominated the scene with neural networks [31,55,56], genetic algorithms [57–82], clustering strategies [83], ant colony optimization [76,84–89], firefly algorithm [89], ant colony system [90,91], market-based algorithms [92,93], imperialist competitive algorithm [94], tabu search [56], gravitational emulation local search algorithm [95], variable neighborhood search [96–99], bee colony optimization [100], invasive weed optimization [100], wolf pack search algorithm [101], discrete pigeon optimization [102], reinforcement learning [103], evolutionary strategies [104], hybrid search [105], memetic search [105], simulated annealing [106], and bees algorithm [107]. As in years before, only a few authors worked on DFmTSP. Remarkably, until 2017 and 2021, the first reported IPs for DFmTSP were published [51,104]. From them, we could reproduce and validate the IP of Karabulut et al. [104]. Nevertheless, in Sections 3.1 and 3.2, we use similar ideas to those mentioned by Assaf [51].

2.6. *Approximation Algorithms*

Only a few approximation algorithms have been designed for some variants of mTSP. For that reason, this paragraph presents an independent account of such algorithms; these include a $(4/3)$- and a $(3/2)$-approximation algorithm for the SmTSP and MmTSP on a tree with two salespersons ($m = 2$) and minmax objective function [108], a $(2 - 2/(m+1))$-approximation algorithm for DFmTSP on trees with m traveling salespersons and minmax objective function [109], a faster approximation algorithm with the same approximation factor for the same problem [110], a 2-approximation algorithm for MmTSP with triangle inequality [111], a $(3/2)$-approximation algorithm for MmTSP with triangle inequality and a constant number of depots [112], a $(2 - 1/k)$-approximation algorithm for MmTSP [113], a $(2 - 1/(2k))$-approximation algorithm for MmTSP [114], a $(1 + \epsilon)$-approximation algorithm for SmTSP on a tree with minmax objective function, with the depot located at the tree's root [115], and a $(1 + \epsilon)$-approximation algorithm for the SmTSP on a spider, with the depot located at its center [116].

2.7. *When Depots Are Unknown or Unnecessary*

Table 1 shows the main scope of the examined papers, and Figure 2 shows how they distribute over time. To collect these papers, we manually tracked the connected citation network of papers directly related to mTSP. From Table 1 and Figure 2, we can observe that, on the one hand, SmTSP and MmTSP have been the most studied variants. On the other hand, DFmTSP has not received as much attention; only two IPs have been published, and they are relatively recent and have a limited scope [51,104]. Since the classical TSP does not require the depot concept to be formulated, we believe that DFmTSP should be considered an essential member of the mTSP collection and should receive more attention. Additionally, this variant is more adequate for specific applications, such as submarine patrol routing [36], supervisor allocation [50,51], and some variations of the job scheduling problem [32]. In a nutshell, DFmTSP is a better model for social problems where depots are unknown or unnecessary. The following sections introduce novel IPs for DFmTSP and its main variants:

- Closed paths (CP).
- Open paths (OP).
- Minsum objective function.
- Minmax objective function.
- Bounding constraints:
  - Lower bound on the number of vertices per path.
  - Upper bound on the number of vertices per path.

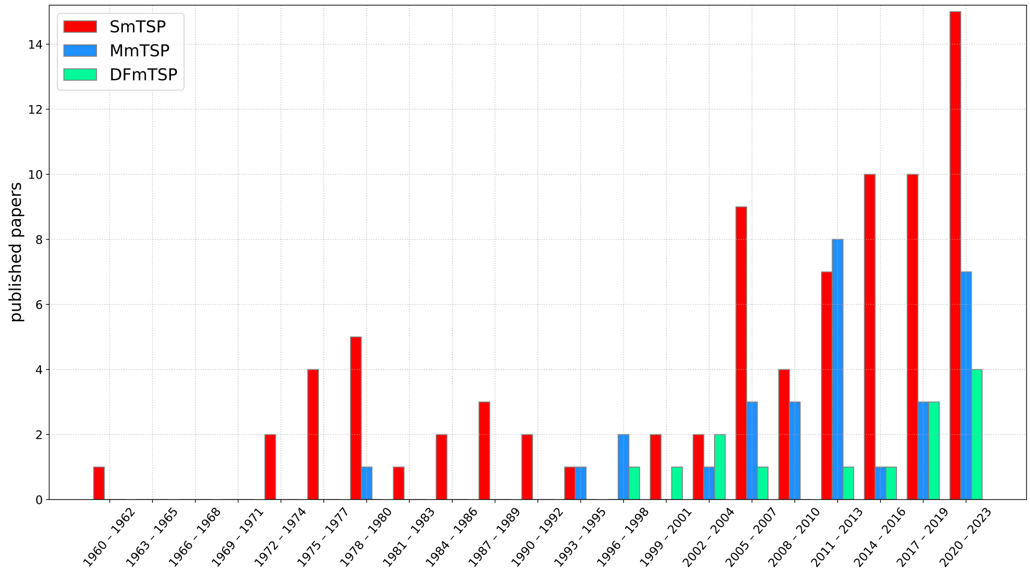


Figure 2. Published papers directly related to mTSP over time.

Table 1. Main categories and scope of related work. S, M, and DF stand for single depot, multiple depot, and depot free, respectively.

Main Scope	mTSP	Reference
Integer programming	S	[4–6,10–22,49]
	M	[6,15,18,42,44,47,48,50–54]
	DF	[36,51,104]
Exact algorithm	S	[14,20,23,45,46]
	M	[42–45,47,115]
	DF	-
Heuristic	S	[19,24–27,40,41]
	M	[38–40]
	DF	[37,40]
Metaheuristic	S	[28–31,34,35,55–65,67–74,76,78,80,81] [83–85,87,88,90,91,94–98,100–105,107,117]
	M	[82,86,88,89,92,93,99,104,105,118]
	DF	[32,33,66,75,77,79,104,106]
Approximation algorithm	S	[108,116]
	M	[108,110–115]
	DF	[109]

As a byproduct, the proposed IPs are adapted to a combination of FD-MmTSP and DFmTSP, namely, a variant where fewer than m depots are part of the input, but the solution consists of exactly m paths. This variant can be helpful in situations where only a few depots have already been selected. Thus, deciding the location of the remaining depots is part of the problem.

3. Integer Programs for DFmTSP

This section introduces novel integer programs (IPs) for DFmTSP and its main variants; one of the main attributes of these IPs is the presence of dummy depots. This section is divided into two parts. Section 3.1 introduces IPs for the CP and OP variants with bounding constraints, and minsum and minmax objective functions. Such IPs have $\mathcal{O}(n^2m)$ binary variables and are based on an IP from the literature for FD-MmTSP [7]. Section 3.2 introduces more compact integer programs for the CP and OP variants with bounding constraints, and minsum objective function. Such IPs have $\mathcal{O}(n^2)$ binary variables.

Sections 3.1 and 3.2 begin by introducing integer quadratic programs (IQPs) for the CP-DFmTSP with a minsum objective function. Afterward, such IQPs are linearized and extended to other variants of the problem, including bounding constraints, OP, and the minmax objective function in the case of Section 3.1.

3.1. Based on FD-MmTSP

The main attribute of the proposed IPs is adding vertices to the input graph, which has become a common practice [5,10–12]. The added vertices are usually of two types: exact copies of an actual depot (part of the input) or dummy depots.

Definition 1. *A dummy depot is a vertex $v_k \notin V(G)$, such that $\forall v_i \in V(G), c_{i,k} = c_{k,i} = 0$, where G is the input graph and $c_{a,b}$ is the cost of the edge (v_a, v_b). As the name suggests, it plays the role of a fake depot.*

This section's IPs extend a state-of-the-art IP for FD-MmTSP [7]. The connection between this problem and DFmTSP comes from the following intuitive observation.

Observation 1. *A solution to CP-DFmTSP defines a partition of vertices. If one vertex from each partition's element is known in advance, the problem can be directly modeled as CP-FD-MmTSP.*

Expressions (1)–(13) show an IP for minsum CP-DFmTSP with bounding constraints. We will use this IP as the basis for the rest of this section's IPs. Namely, we will show how to adapt this formulation to the minmax objective function and the OP variant. So, let us begin by explaining this IP. V is the set of vertices of the weighted complete graph $G = (V, E)$, and D is a set of m dummy depots. To be more general, let us consider the input graph to be directed. For clarity, let us assume that the vertices in V are labeled as $\{v_1, v_2, ..., v_n\}$ and the dummy depots in D as $\{v_{n+1}, v_{n+2}, ..., v_{n+m}\}$ ($V \cap D = \emptyset$). $c_{i,j}$ is the cost of traveling from v_i to v_j. Notice that $c_{i,j}$ might be different from $c_{j,i}$; namely, we are working with the asymmetric version of the problem, which generalizes the symmetric one.

$$
\begin{aligned}
\min \quad & \sum_{v_i\in V}\sum_{v_j\in V} c_{i,j} \sum_{v_k\in D}\left(x_{i,j,k} + x_{i,k,k}x_{k,j,k}\right) && && (1)\\
\text{s.t.} \quad & \sum_{v_j\in V} x_{k,j,k} = 1 && \forall v_k \in D && (2)\\
& \sum_{v_k\in D} x_{k,j,k} + \sum_{v_k\in D}\sum_{v_i\in V} x_{i,j,k} = 1 && \forall v_j \in V && (3)\\
& x_{k,j,k} + \sum_{v_i\in V} x_{i,j,k} - x_{j,k,k} - \sum_{v_i\in V} x_{j,i,k} = 0 && \forall v_k \in D, v_j \in V && (4)\\
& \sum_{v_j\in V} x_{k,j,k} - \sum_{v_j\in V} x_{j,k,k} = 0 && \forall v_k \in D && (5)\\
& t_i + (U-2)\sum_{v_k\in D} x_{k,i,k} - \sum_{v_k\in D} x_{i,k,k} \le U-1 && \forall v_i \in V && (6)\\
& t_i + \sum_{v_k\in D} x_{k,i,k} + (2-L)\sum_{v_k\in D} x_{i,k,k} \ge 2 && \forall v_i \in V && (7)\\
& \sum_{v_k\in D} x_{k,i,k} + \sum_{v_k\in D} x_{i,k,k} \le 1 && \forall v_i \in V && (8)\\
& t_i - t_j + U\sum_{v_k\in D} x_{i,j,k} + (U-2)\sum_{v_k\in D} x_{j,i,k} \le U-1 && \forall v_i, v_j \in V && (9)\\
& x_{i,j,k} \in \{0,1\} && \forall v_i, v_j \in V\cup D, \forall v_k \in D && (10)
\end{aligned}
$$

where

$$
x_{i,j,k} = \begin{cases} 1, & \text{if the salesperson at the } v_k \text{ dummy depot goes from } v_i \text{ to } v_j \\ 0, & \text{otherwise} \end{cases} \tag{11}
$$

$$
t_i = \text{time at which vertex } v_i \text{ is visited in the path} \tag{12}
$$

$$
2 \le L \le \lceil |V|/m \rceil \le U \le |V| \tag{13}
$$

In this formulation, constraints (2) ensure that precisely one salesperson departs from each dummy depot $v_k \in D$. Constraints (3) ensure that each vertex $v_j \in V$ is visited once from some vertex $v_i \in V \cup D$. Constraints (4) and (5) guarantee the route continuity for vertices and dummy depots. Constraints (6), (7), and (13) guarantee that each salesperson visits between L (lower-bound) and U (upper-bound) vertices; we refer to these as bounding constraints. Constraints (8) forbid a salesperson to visit only one vertex (also known as a return trip). Constraints (9) are subtour elimination constraints (SECs) that ensure $t_j = t_i + 1$ if and only if $x_{i,j} = 1$. Constraints (10) define the decision variables. Notice that we use the word time metaphorically; variables t represent the order in which a salesperson visits vertices. Finally, the objective function (1) guarantees that the total paths' cost is minimized (minsum). The number of binary variables and constraints is $\mathcal{O}(n^2 m)$ and $\mathcal{O}(n^2)$, respectively. Notice that the objective function has a quadratic term. Therefore, this program is an integer quadratic program (IQP); however, as we will show later, this can be linearized by adding some extra variables and constraints.

Before continuing with abstract statements, Figure 3 shows the optimal solution of this IP over a small graph with $m = 2$, $L = 3$, and $U = 5$. In this figure, two salespersons depart and return to the dummy depots v_9 and v_{10}, respectively. Since traveling through a dummy depot does not incur any cost, we have to identify the edges (v_2, v_1) and (v_6, v_5) and add their cost to the objective function. This is explained in Lemma 1. Regarding matrices $\mathbf{x}_9$ and $\mathbf{x}_{10}$, each corresponds to a different dummy depot and codifies a different closed path.

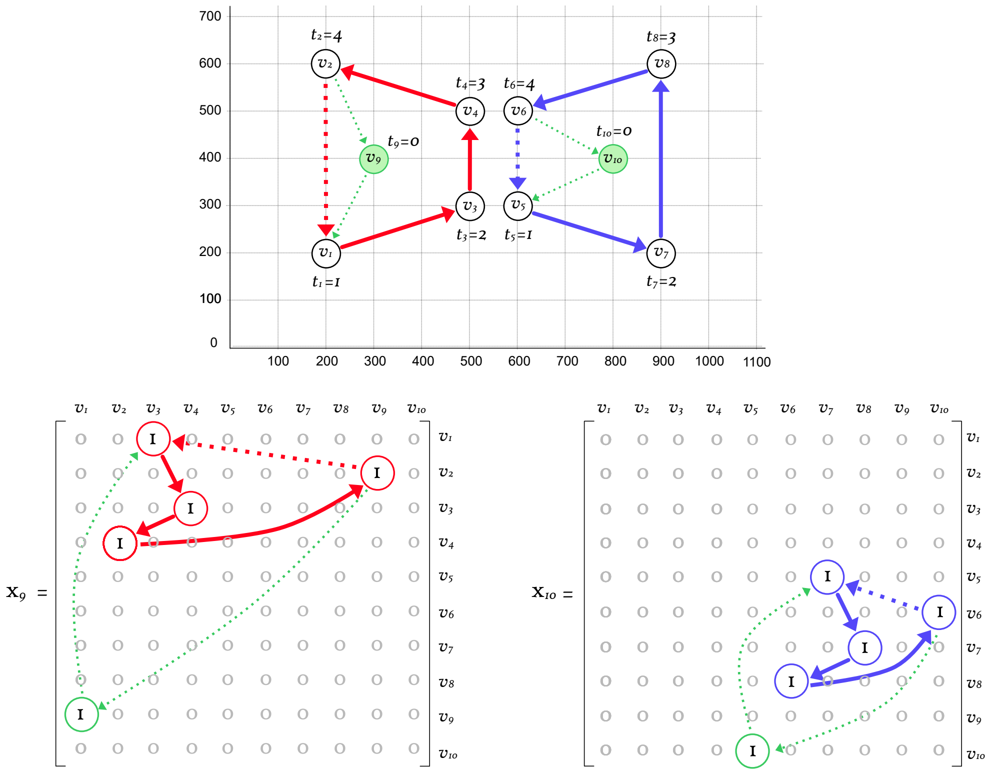


Figure 3. Exact solution of the IP for minsum CP-DFmTSP with bounding constraints. In this graph instance, $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$, $m = 2$, $L = 3$, $U = 5$, $D = \{v_9, v_{10}\}$, and the cost of each edge equals the euclidean distance between its vertices (except edges with some dummy depot). There is a 10×10 matrix for each dummy depot, $\mathbf{x}_9 = [x_{i,j}]_{10\times 10}$ and $\mathbf{x}_{10} = [x_{i,j}]_{10\times 10}$.

Lemma 1. *Expression (1) is the minsum objective function for CP-DFmTSP.*

Proof. Expression (1) comes from the following expanded form:

$$\underbrace{\sum_{v_k \in D} \sum_{v_i \in V} \sum_{v_j \in V} c_{i,j} x_{i,j,k}}_{\text{first term}} + \underbrace{\sum_{v_k \in D} \sum_{v_i \in V} \sum_{v_j \in V} c_{i,j} x_{i,k,k} x_{k,j,k}}_{\text{second term}} \tag{14}$$

The first term of Expression (14) adds up the cost of the traveled edges (solid red and blue edges in Figure 3). The second term adds up the cost $c_{i,j}$ of the edge of each pair of vertices $v_i, v_j \in V$ adjacent to a dummy depot $v_k \in D$ (dotted red and blue edges in Figure 3). Let v_k be any dummy depot in D, and let v_i and v_j be any pair of different vertices in V. If $x_{i,k,k} = x_{k,j,k} = 1$, then the salesperson associated with the v_k dummy depot goes from vertex v_i to v_k and then from vertex v_k to v_j. Thus, the path is closed if we consider the edge (v_i, v_j). This way, the cost of edge (v_i, v_j) is considered in the objective function because $c_{i,j} x_{i,k,k} x_{k,j,k} = c_{i,j}$. Finally, the Objective function (14) considers the sum of the paths of all the salespersons. Therefore, it is the minsum objective function for CP-DFmTSP. □

Although the objective function (1) has a quadratic term, it can be linearized by noting that, in the second term of Expression (14), $c_{i,j}$ is added up if and only if $x_{i,k,k} x_{k,j,k} = 1$. So, this product can be replaced by the binary variable $y_{i,j,k}$ (see constraints (16)–(19)). By doing so, the minsum objective function is Equation (15) and we obtain an integer linear program (ILP) with $\mathcal{O}(n^2 m)$ binary variables and constraints.

$$\sum_{v_i \in V} \sum_{v_j \in V} c_{i,j} \sum_{v_k \in D} \left(x_{i,j,k} + y_{i,j,k} \right) \tag{15}$$

where

$$y_{i,j,k} \geq x_{i,k,k} + x_{k,j,k} - 1 \qquad \forall v_i, v_j \in V, \forall v_k \in D \tag{16}$$
$$y_{i,j,k} \leq x_{k,j,k} \qquad \forall v_i, v_j \in V, \forall v_k \in D \tag{17}$$
$$y_{i,j,k} \leq x_{i,k,k} \qquad \forall v_i, v_j \in V, \forall v_k \in D \tag{18}$$
$$y_{i,j,k} \in \{0,1\} \qquad \forall v_i, v_j \in V, \forall v_k \in D \tag{19}$$

The IP for minsum CP-DFmTSP can be adapted for the minmax objective function if we replace Expression (1) with Expressions (20) and (21), where P_{max} is the longest path among the salespersons.

$$\min \quad P_{max} \tag{20}$$
$$P_{max} \geq \sum_{v_i \in V} \sum_{v_j \in V} c_{i,j} \left(x_{i,j,k} + x_{i,k,k} x_{k,j,k} \right) \qquad \forall v_k \in D \tag{21}$$

Lemma 2. *Expressions (20) and (21) are the minmax objective function for CP-DFmTSP.*

Proof. Expression (21) comes from the following expanded form:

$$P_{max} \geq \underbrace{\sum_{v_i \in V} \sum_{v_j \in V} c_{i,j} x_{i,j,k}}_{\text{first term}} + \underbrace{\sum_{v_i \in V} \sum_{v_j \in V} c_{i,j} x_{i,k,k} x_{k,j,k}}_{\text{second term}} \qquad \forall v_k \in D \tag{22}$$

Let $v_k \in D$ be a specific dummy depot and let $v_i, v_j \in V$ be a pair of vertices in its respective path. Then, the first term of Expression (22) adds up the cost of the edges in the path of the dummy depot v_k, and the second term adds the cost $c_{i,j}$ of the edge (v_i, v_j) such that the salesperson goes from v_i to v_k and then from v_k to v_j. Thus, by considering the edge (v_i, v_j) the path is closed; the cost of this edge is considered in the objective function because $c_{i,j} x_{i,k,k} x_{k,j,k} = c_{i,j}$. Note that Expression (22) computes the cost of the closed path per each dummy depot $v_k \in D$. Additionally, variable P_{max} must be greater or equal to each salesperson's path's cost and must be minimized. So, Expressions (20) and (21) are the minmax objective function for CP-DFmTSP. □

As with minsum, the minmax objective function defined by Expressions (20) and (21) can be linearized by adding variables $y_{i,j,k} = x_{i,k,k} x_{k,j,k}$ and constraints (16)–(18).

In this paper, a singleton path (v_i) of length 0 is invalid. However, a two-vertices path (v_i, v_j) is a valid closed path of length $c_{i,j} + c_{j,i}$ or an open path of length $c_{i,j}$. Therefore, a solution to any variant of DFmTSP must consist of m paths, each with at least two vertices. Of course, the number of vertices in each path must also be between the bounding constraints L and U. Lemmas 3 and 4 show that the proposed IPs are consistent with these considerations.

Lemma 3. *Constraints (9) guarantee that $t_b = t_a + 1$ if and only if $x_{a,b,k} = 1$ for some dummy depot v_k [7].*

Proof. First, let us consider the case of a pair of vertices, v_a and v_b, such that $x_{a,b,k} = x_{b,a,k} = 0$ for every dummy depot v_k. In this scenario, constraints (9) are reduced to the single constraint $t_a - t_b \leq U - 1$, which is basically deactivated. Next, let us consider the case of a pair of vertices v_a and v_b such that $x_{a,b,k} = 1$ and $x_{b,a,k} = 0$ for some dummy depot v_k. In this scenario, constraints (9) become two constraints, $t_j \geq t_i + 1$ (with $i = a$ and $j = b$) and $t_i \leq t_j + 1$ (with $i = b$ and $j = a$). Thus, $t_a + 1 \leq t_b \leq t_a + 1$, i.e., $t_b = t_a + 1$. □

Lemma 4. *A solution to the proposed IPs consists of m paths, each with between* $\max\{2, L\}$ *and* U *vertices.*

Proof. By constraints (8), $\forall v_i \in V$, $\sum_{v_k \in D} x_{k,i,k} = \sum_{v_k \in D} x_{i,k,k} = 1$ is not allowed; this case alone avoids singleton paths from occurring. Now, let us inspect the three remaining cases; the consequent of each implication follows from constraints (6) and (7) combined:

1. If $\sum_{v_k \in D} x_{k,i,k} = \sum_{v_k \in D} x_{i,k,k} = 0$, then $2 \le t_i \le U - 1$.
   - This case corresponds to the vertices non-adjacent to any dummy depot in any path. In Figure 3, these are v_3, v_4, v_7, and v_8.
2. If $\sum_{v_k \in D} x_{k,i,k} = 1$ and $\sum_{v_k \in D} x_{i,k,k} = 0$, then $t_i = 1$.
   - This case corresponds to the first vertex visited by each salesperson after leaving its dummy depot. In Figure 3, these are v_1 and v_3.
3. If $\sum_{v_k \in D} x_{k,i,k} = 0$ and $\sum_{v_k \in D} x_{i,k,k} = 1$, then $L \le t_i \le U$.
   - This case corresponds to the last vertex visited by each salesperson before returning to its dummy depot. In Figure 3, these are v_2 and v_6.

By Lemma 3, $t_j = t_i + 1$ if and only if $x_{i,j} = 1$. In other words, the variables t_i and t_j of adjacent vertices v_i and v_j in a path must differ by exactly one unit. Thanks to this, constraints (6) and (7) guarantee that each path has between $\max\{2, L\}$ and U vertices. Finally, since there cannot be empty paths, a solution must have exactly m paths. □

So far, we have introduced and explained the main elements of IPs for CP-DF*m*TSP with bounding constraints and minsum and minmax objective functions. Next, we adapt this formulation to the OP variant. To our knowledge, this is the first reported mathematical formulation for this variant. Fortunately, all we have to modify is the objective function. The minsum objective function for the OP variant is:

$$\sum_{v_i \in V} \sum_{v_j \in V} c_{i,j} \left(\sum_{v_k \in D} x_{i,j,k} \right) \tag{23}$$

Lemma 5. *Expression (23) is the minsum objective function for OP-DFmTSP.*

Proof. Expression (23) is the first term of expression (14). It considers only the cost of the traveled edges (solid red and blue edges in Figure 3). The vertices v_i and v_j that make $x_{i,k,k} = 1$ and $x_{k,j,k} = 1$ are in the extremes of the open path; thus, $c_{i,j}$ needs not be considered. Finally, the objective function (23) considers the sum of the path's length for all salespersons. Therefore, it is the minsum objective function for OP-DF*m*TSP. □

Next, we show the minmax objective function for the OP variant.

$$\min \quad P_{max} \tag{24}$$

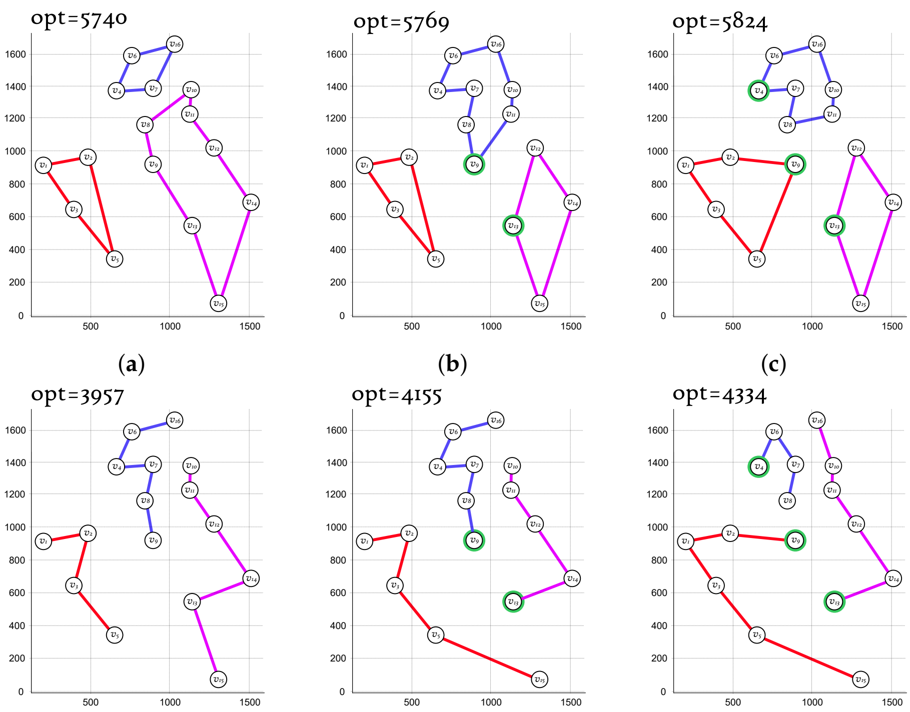
$$P_{max} \ge \sum_{v_i \in V} \sum_{v_j \in V} c_{i,j} x_{i,j,k} \qquad \forall v_k \in D \tag{25}$$

Lemma 6. *Expressions (24) and (25) are the minmax objective function for OP-DFmTSP.*

Proof. Expression (25) is the first term of Expression (22). It only considers the cost of the traveled edges. So, the paths are open. Additionally, variable P_{max} is greater or equal to each salesperson's open path length, and P_{max} is minimized. Therefore, Expressions (24) and (25) are the minmax objective function for OP-DF*m*TSP. □

By Lemmas 2–4, the proposed IPs are valid formulations for DF*m*TSP and its main variants: OP, CP, minsum, minmax, and bounding constraints. To finish this section, notice that all these IPs can be adapted to FD-M*m*TSP by adding constraints (26); these force each dummy depot to directly visit a vertex $v \in R$, where R is the input set of depots, $V \cap R = R$, $V \cap D = \emptyset$, and $|R| = m$. Of course, this adaptation is redundant because, in the first place, the proposed IPs extend an IP for FD-MmTSP. However, this adaption becomes useful when $|R| < m$; in this scenario, we deal with a combination between FD-MmTSP and DFmTSP where fewer than m depots are known, but the solution still consists of m paths. For further illustration, Figure 4 shows the optimal solution for Mexico's cities' town halls graph (cdmx16). We implemented and executed all the proposed formulations to compute the optimal solutions using off-the-shelf optimization software [119] (see the Data Availability Statement.)

$$\sum_{v_k \in D} \sum_{v_i \in R} x_{k,i,k} = |R| \tag{26}$$



(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

Figure 4. Optimal solutions for (**a**) CP-DFmTSP, (**b**) a combination between CP-DFmTSP and CP-FD-MmTSP ($R = \{v_9, v_{13}\}$), and (**c**) CP-FD-MmTSP ($R = \{v_9, v_{13}, v_4\}$). The objective function is minsum, the number of salespersons is three ($m = 3$), $L = 4$, $U = 10$, the cost of each edge equals the euclidean distance between its vertices, and the depots are marked in green. Subfigures (**d**–**f**) correspond to the open-paths (OP) variants.

The following section introduces more compact formulations for the same problems and most of their variants.

3.2. More Compact Programs for DFmTSP

This section introduces more compact IPs for DFmTSP and some of its main variants. The main IP of this section has $\mathcal{O}(n^2)$ binary variables, $\mathcal{O}(n^2)$ constraints, and requires m dummy depots (see Definition 1). Expressions (27)–(37) define an IP for minsum CP-DFmTSP; we will use this IP as the basis for the rest of this section's IPs. For clarity, let us assume that the vertices in V are labeled as $\{v_1, v_2, ..., v_n\}$, the m dummy depots in D are labeled as $\{v_{n+1}, v_{n+2}, ..., v_{n+m}\}$, and $V' = V \cup D$, where $G = (V, E)$ is the input graph. Notice that $V \cap D = \emptyset$.

$$
\begin{aligned}
\min \quad & \sum_{v_i \in V'} \sum_{v_j \in V'} c_{i,j} \Big(x_{i,j} + x_{n+m,j} x_{i,n+1} \\
& \qquad + \sum_{v_k \in D \setminus \{v_{n+m}\}} x_{k,j} x_{i,k+1} \Big) && (27) \\
\text{s.t.} \quad & \sum_{v_j \in V' \setminus \{v_i\}} x_{i,j} = 1 && \forall v_i \in V' \quad (28) \\
& \sum_{v_i \in V' \setminus \{v_j\}} x_{i,j} = 1 && \forall v_j \in V' \quad (29) \\
& t_{n+1} = 0 && (30) \\
& t_{k+1} - t_k \geq 3 && \forall v_k \in D \setminus \{v_{n+m}\} \quad (31) \\
& |V'| - t_{n+m} \geq 3 && (32) \\
& 1 \leq t_i \leq |V'| - 1 && \forall v_i \in V' \setminus \{v_{n+1}\} \quad (33) \\
& t_i - t_j + x_{i,j}|V'| \leq |V'| - 1 && \forall v_i, v_j \in V' \setminus \{v_{n+1}\} \quad (34) \\
& x_{i,j} \in \{0,1\} && \forall v_i, v_j \in V' \quad (35)
\end{aligned}
$$

where

$$
x_{i,j} = \begin{cases} 1, & \text{if a salesperson travels from } v_i \text{ to } v_j \\ 0, & \text{otherwise} \end{cases} \tag{36}
$$

$$
t_i = \text{time at which vertex } v_i \text{ is visited in the path} \tag{37}
$$

The objective function (27) has a quadratic term. Therefore, this formulation is an IQP. Later, we will show how to linearize the objective function; but before, let us explain this formulation. The solution to this formulation is a closed path that visits all vertices once, similar to TSP. From this single path, all m paths are inferred. Constraints (28) and (29) are flow constraints; they guarantee that there is a single closed path that visits all vertices once. Constraints (30)–(32) are depot ordering constraints; they have two goals, to avoid singleton paths and to force the salespersons to visit the dummy depots in order, i.e., $t_{n+1} < t_{n+2} < \cdots < t_{n+m}$. Constraints (33) and (34) are the classical Miller–Tucker–Zemlin SECs [4]. Expressions (35)–(37) define the decision variables. The objective function (27) is minsum (see Lemma 7). Notice that this IP has only one bounding constraint, i.e., each path must have more than two vertices. However, to extend this IP to the more general case, we can add constraints (38)–(42) and remove constraints (31) and (32). In this manner, each path must have between L and U vertices.

$$
\begin{aligned}
& t_{k+1} - t_k \leq U + 1 && \forall v_k \in D \setminus \{v_{n+m}\} \quad (38) \\
& t_{k+1} - t_k \geq L + 1 && \forall v_k \in D \setminus \{v_{n+m}\} \quad (39) \\
& |V'| - t_{n+m} \leq U + 1 && (40) \\
& |V'| - t_{n+m} \geq L + 1 && (41) \\
& 2 \leq L \leq \lceil |V|/m \rceil \leq U \leq |V| && (42)
\end{aligned}
$$

Figure 5 shows the optimal solution of this IP over a small graph with $m = 2$, $L = 3$, and $U = 5$. In this figure, two paths are codified into one path that departs from dummy depot v_9, travels through every other vertex, including dummy depot v_{10}, and returns to v_9. Since traveling through a dummy depot does not incur any cost, we must identify the edges (v_1, v_2) and (v_8, v_7) and add their cost to the objective function. This is explained in Lemma 7. Notice that one matrix is enough to codify all m paths.

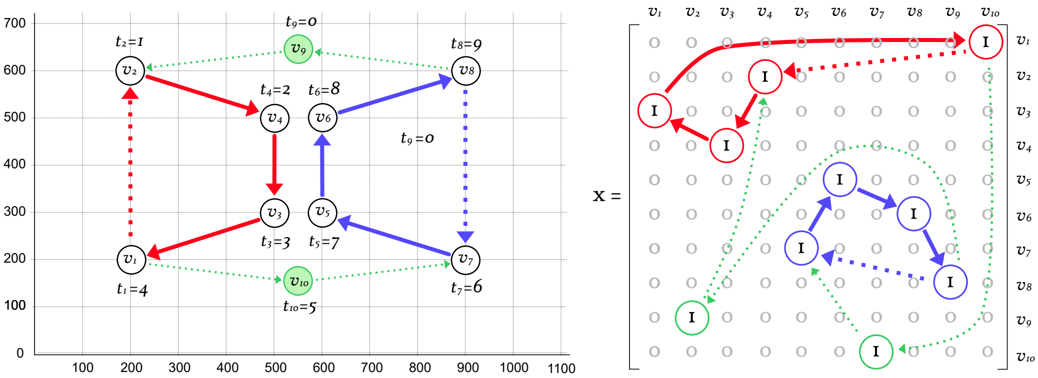


Figure 5. Exact solution of the IP for minsum CP-DFmTSP with bounding constraints. In this graph instance, $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$, $m = 2$, $L = 3$, $U = 5$, $D = \{v_9, v_{10}\}$, and the cost of each edge equals the euclidean distance between its vertices (except edges with some dummy depot). There is only one matrix, $\mathbf{x} = [x_{i,j}]_{10\times 10}$.

Lemma 7. *Expression (27) is the minsum objective function for CP-DFmTSP.*

Proof. Expression (27) comes from the following expanded form:

$$\underbrace{\sum_{v_i\in V'}\sum_{v_j\in V'} c_{i,j}x_{i,j}}_{\text{first term}} + \underbrace{\sum_{v_i\in V'}\sum_{v_j\in V'} c_{i,j}\left(x_{n+m,j}x_{i,n+1} + \sum_{v_k\in D\setminus\{v_{n+m}\}} x_{k,j}x_{i,k+1}\right)}_{\text{second term}} \tag{43}$$

The first term of Expression (43) adds up the cost of the traveled edges (solid red and blue edges in Figure 5). The second term adds up the cost $c_{i,j}$ of the edge of each pair of vertices $v_i, v_j \in V$ adjacent to a pair of consecutive dummy depots (dotted red and blue edges in Figure 5). Let v_k and v_{k+1} be a pair of consecutive dummy depots in $D \setminus \{v_{n+m}\}$ and let v_i and v_j be any pair of different vertices in V. If $x_{k,j} = 1$ and $x_{i,k+1} = 1$, then the salesperson associated with the v_k dummy depot goes from vertex v_i to v_j (notice that v_{n+1} is the consecutive dummy depot of v_{n+m}.) Thus, the path is closed if we consider the edge (v_i, v_j). Thanks to the flow constraints (28) and (29), and the depot ordering constraints (30)–(32), the inner sum $x_{n+m,j}x_{i,n+1} + \sum_{v_k\in D\setminus\{v_{n+m}\}} x_{k,j}x_{i,k+1}$ can only take values in $\{0,1\}$. Therefore, Expression (27) is the minsum objective function for CP-DFmTSP. □

Although the objective function (27) has a quadratic term, it can be linearized by noting that, in the second term of Expression (43), $c_{i,j}$ is added up if and only if there is a pair of consecutive dummy depots, v_k and v_{k+1}, such that $x_{k,j}x_{i,k+1} = 1$. So, this product can be replaced by the binary variable $y_{i,j}$ (see constraints (45)–(49)). By doing so, the minsum objective function is Equation (44) and we obtain an ILP with $\mathcal{O}(n^2)$ binary variables and $\mathcal{O}(n^2 m)$ constraints. Notice that v_{n+1} is the consecutive dummy depot of v_{n+m}.

$$\sum_{v_i\in V'}\sum_{v_j\in V'} c_{i,j}(x_{i,j} + y_{i,j}) \tag{44}$$

where

$$y_{i,j} \geq x_{k,j} + x_{i,k+1} - 1 \qquad \forall v_i, v_j \in V, \forall v_k \in D \setminus \{v_{n+m}\} \tag{45}$$

$$y_{i,j} \geq x_{n+m,j} + x_{i,n+1} - 1 \qquad \forall v_i, v_j \in V \tag{46}$$

$$y_{i,j} \leq \sum_{v_k \in D} x_{i,k} \qquad \forall v_i, v_j \in V \tag{47}$$

$$y_{i,j} \leq \sum_{v_k \in D} x_{k,j} \qquad \forall v_i, v_j \in V \tag{48}$$

$$y_{i,j} \in \{0,1\} \qquad \forall v_i, v_j \in V \tag{49}$$

Adapting these IPs to the OP variant is straightforward. We only have to replace the objective function (27) by (50), and omit variables $y_{i,j}$ with their respective constraints. This way, the dotted lines from Figure 5 are not considered.

$$\sum_{v_i \in V'} \sum_{v_j \in V'} c_{i,j} x_{i,j} \tag{50}$$

To finish this section, notice that this section's IPs can be adapted to FD-MmTSP with CP, OP, minsum, and bounding constraints. We must include constraints (51), which force each dummy depot to go directly to one depot from the input set R of actual depots. Figure 6 serves as an example for $R = \{v_2, v_4\}$. As the previous section's IPs, this adaptation is advantageous too when $|R| < m$ (see Figure 4).

$$\sum_{v_k \in D} x_{k,i} = 1 \qquad \forall v_i \in R \tag{51}$$

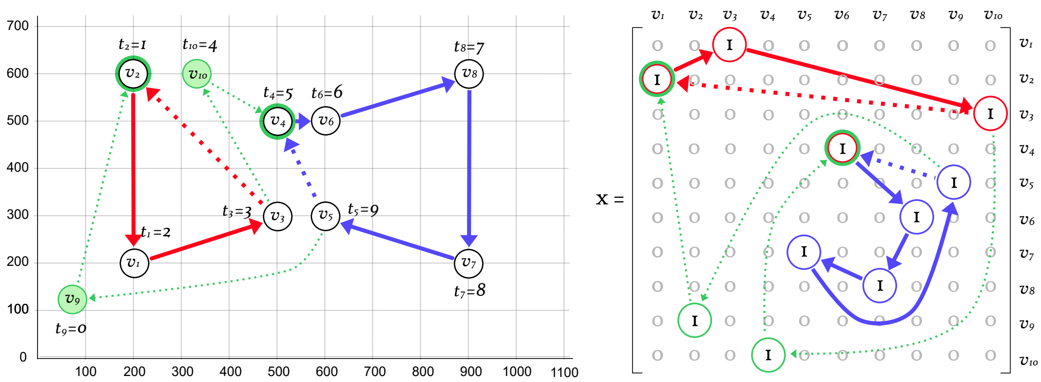


Figure 6. Exact solution of the IP for minsum CP-FD-MmTSP with bounding constraints (Section 3.2). In this graph instance, $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$, $m = 2$, $L = 3$, $U = 5$, $R = \{v_2, v_4\}$, $D = \{v_9, v_{10}\}$, and the cost of each edge equals the euclidean distance between its vertices (except edges with some dummy depot). There is only one matrix, $\mathbf{x} = [x_{i,j}]_{10 \times 10}$.

4. Empirical Tests

To test the empirical performance of the proposed IPs, we ran some experiments using off-the-shelf optimization software and some of the classical TSPLIB graph instances [120]. From the literature, we could reproduce and validate the IP of Karabulut et al. [104]; we used this IP for comparison purposes. In all the experiments, we set the lower bounding constraint to two ($L = 2$) because the IP of Karabulut et al. is limited to such a value.

All the IPs were implemented using Gurobi 9.5.1 (Gurobi, Beaverton, OR, USA) [119] through its Python interface. This off-the-shelf optimization software executes optimization techniques such as simplex, branch and bound, branch and cut, cutting planes, parallelism, and heuristics to find optimal solutions to mathematical formulations with linear or quadratic constraints. In all the experiments, we set Gurobi's Presolve parameter to 2. All the experiments were executed on a desktop Windows 11 Pro computer with an Intel Core i7-9700 processor. All the executions were set to a maximum of 12 GB of RAM and a 2 h time limit.

Tables 2 and 3 show the results obtained by all the proposed IPs and the IP of Karabulut et al. [104] for the minsum CP-DFmTSP. We only experimented with this problem variant because it is the only one modeled by all the presented IPs, including the IP of Karabulut et al. In these tables, f stands for the reported value of the objective function, $t(s)$ for the time in seconds to find such objective function value, and g for the gap reported by the Gurobi software. A dash "-" character means no feasible solution was found within two hours. For convenience, we refer to the IPs from Section 3.1 with quadratic objective function as IQP1, and to the IPs from the same section with linear objective function as ILP1. Similarly, we refer to the IPs from Section 3.2 as IQP2 and ILP2.

Table 2 shows the results for the minsum CP-DFmTSP with tight bounding constraints, namely, the upper bound U is set to $\lceil n/m \rceil$. On the one hand, we can observe that only IQP1 and ILP2 could find feasible solutions for all the cases, suggesting they may be better suited for this specific variant of the problem. Additionally, IQP1 found the best solutions in more cases than others. On the other hand, the IP of Karabulut et al. could not find feasible solutions in three cases within the two-hour time limit. Figure 7 shows the convergence of the IPs reported by the Gurobi software for one of the used instances. From this figure, we can observe two facts. Firstly, from Figure 7a ($m = 3$), we can see that for some IPs it may not make sense to let them run for a long time since most of the IPs reach their best-found solution in much less time than the limit setup. Secondly, from Figure 7b ($m = 5$), we observe cases where the IPs may require more time to find feasible solutions. For instance, IQP1, IQP2, ILP1, and ILP2 needed 546 s, 1565 s, 2209 s, and 3070 s, respectively, to find the first feasible solution (the IP of Karabulut et al. did not find any feasible solution.) These figures suggest that, as expected, the problems with a bigger value of m may be considerably challenging. Similar results were obtained for the other graph instances (see Appendix A).

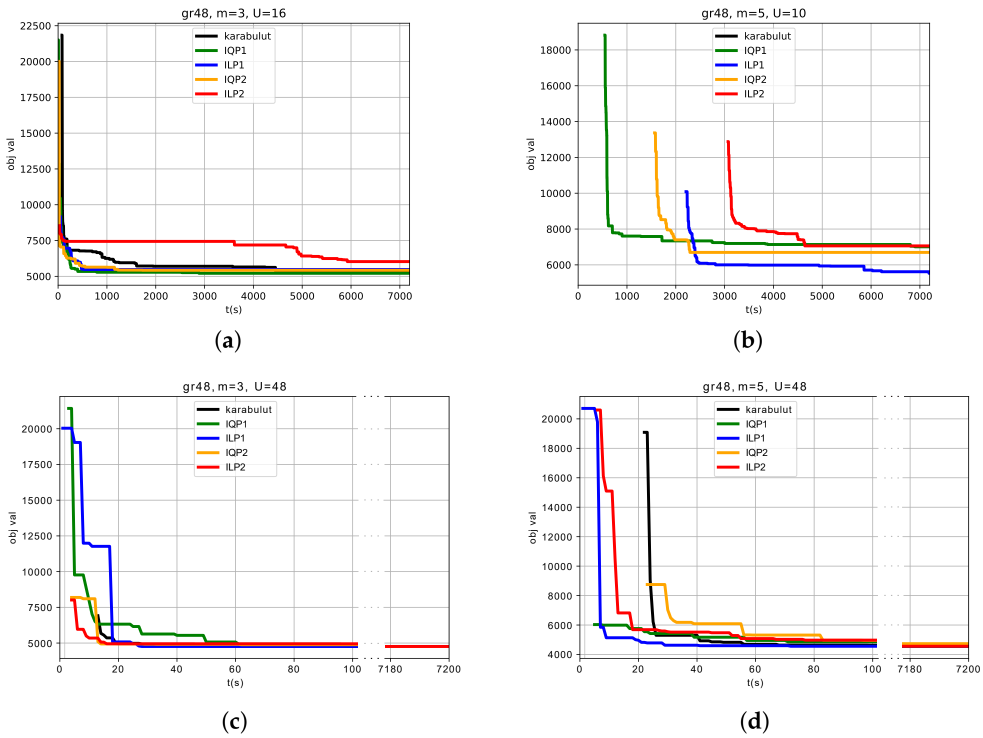


Figure 7. Convergence time reported by Gurobi for the instance gr48 for the minsum CP-DFmTSP with $L = 2$. Subfigures (**a**,**b**) correspond to tight bounding constraints, i.e., $U = \lceil n/m \rceil$. Subfigures (**c**,**d**) correspond to loose bounding constraints, i.e., $U = n$.

Table 2. IPs' comparison for minsum CP-DFmTSP with $L = 2$ and tight bounding constraints. The best-found solutions are bold.

Instance	n	m	U	Karabulut et al. [104]			IQP1			ILP1			IQP2			ILP2		
				f	$t(s)$	g	f	$t(s)$	g	f	$t(s)$	g	f	$t(s)$	g	f	$t(s)$	g
dantzig42	42	3	14	748	320	18%	772	2914	24%	**739**	3934	21%	772	1125	24%	**739**	6823	21%
		5	9	**754**	5559	21%	787	1843	34%	786	2639	34%	879	4030	43%	1055	1763	52%
swiss42	42	3	14	1410	5682	17%	**1407**	1155	21%	1629	6377	33%	1430	1043	24%	1616	107	33%
		5	9	1397	886	19%	1603	4673	38%	**1394**	1741	26%	1470	1493	37%	1787	246	48%
att48	48	3	16	12,557	7200	25%	**10,941**	859	17%	12,339	777	26%	12,683	6329	31%	12,432	668	28%
		5	10	-	-	-	**11,149**	2272	27%	11,505	1884	28%	11,748	4602	33%	16,160	7129	52%
gr48	48	3	16	5337	4452	13%	**5213**	2881	15%	5478	528	21%	5423	1724	23%	6030	5921	30%
		5	10	-	-	-	7009	6895	45%	**5530**	7184	30%	6696	2286	44%	7060	4639	47%
hk48	48	3	16	**11,999**	1656	9%	12,620	663	20%	12,568	6956	19%	13,576	643	26%	12,456	4357	20%
		5	10	-	-	-	13,546	5583	33%	-	-	-	-	-	-	15,316	6864	42%

Table 3. IPs' comparison for minsum CP-DFmTSP with $L = 2$ and loose bounding constraints. The best-found solutions are bold.

Instance	n	m	U	Karabulut et al. [104]			IQP1			ILP1			IQP2			ILP2		
				f	$t(s)$	g	f	$t(s)$	g	f	$t(s)$	g	f	$t(s)$	g	f	$t(s)$	g
dantzig42	42	3	42	**633**	74	0%	**633**	206	5.69%	**633**	91	7.11%	**633**	100	3.95%	**633**	40	3.79%
		5	42	**604**	84	0%	**604**	36	4.64%	**604**	38	14.07%	**604**	135	15.07%	605	83	13.22%
swiss42	42	3	42	**1208**	188	0%	**1208**	89	2.15%	**1208**	27	1.57%	**1208**	203	9.11%	**1208**	27	9.27%
		5	42	**1155**	50	1.47%	**1155**	413	5.11%	**1155**	812	5.63%	1167	2200	19.37%	**1155**	60	17.49%
att48	48	3	48	**9946**	442	4.25%	**9946**	11	5.81%	**9946**	85	4.83%	**9946**	51	7.57%	**9946**	26	7.51%
		5	48	**9448**	43	2.64%	**9448**	182	13.52%	**9448**	400	13.79%	**9448**	85	15.81%	**9448**	2000	14.73%
gr48	48	3	48	**4761**	201	1.70%	**4761**	96	3.91%	**4761**	27	6.87%	**4761**	1369	8.36%	**4761**	227	8.25%
		5	48	**4544**	113	0%	**4544**	207	8.78%	**4558**	71	7.79%	4735	171	18.59%	4558	2829	14.26%
hk48	48	3	48	**11,101**	115	0%	**11,101**	335	2.08%	**11,101**	206	2.20%	11,332	3037	10.98%	11,134	1950	8.45%
		5	48	**10,834**	164	0%	**10,834**	919	6.98%	**10,834**	786	7.30%	10,888	1524	17.20%	10,967	127	17.42%

The experimentation reported in Table 3 is similar to that presented in Table 2. The only difference is that in Table 3, the upper bound constraint U is set to n. From this table, we observe that all of the IPs found feasible solutions for all of the cases, and the IP of Karabulut et al., IQP1, and IQP2 found the best solutions among all the IPs; even the optimality of some of them was proved because a value of $g = 0$ was reported. Furthermore, we can see that the reported gaps are considerably lower than those reported in Table 2. This suggests that the CP-DFmTSP variant with bounding constraints may be harder to solve when the number of vertices per salesperson is tight.

Figure 7c,d show the convergence time reported by Gurobi for the presented IPs for one of the used instances with loose bounding constraints. This figure shows that most IPs quickly found good-quality solutions, considering the relatively small reported gaps. This suggests that for this variant of the problem, setting long running times may not be necessary since the first few seconds (<100 s) are enough to get the most significant improvements in their found solutions. This supports the observation that tight bounding constraints make the instances harder to solve. Similar results were obtained for the other graph instances (see Appendix A).

5. Discussion

Sections 3.1 and 3.2 introduce novel compact integer programs for DFmTSP. These include IQPs and ILPs for the main variants of the problem: CP, OP, with bounding constraints, and minsum and minmax objective functions (see Tables 4 and 5 for more details). Although some other IPs for DFmTSP have been published before [51,104], we could only reproduce and validate the IP of Karabulut et al. [104]. Additionally, these state-of-the-art IPs do not consider all the problem's main variants and are not designed to solve a combination of FD-MmTSP and DFmTSP. To our knowledge, the IPs introduced in this paper are the first reported with such a wide scope.

Section 3.2 introduces more compact IPs for DFmTSP. Specifically, these have $\mathcal{O}(n^2)$ binary variables (see Table 4). These IPs depend on the dummy depot concept, on what we call depot ordering constraints, and they consider most of the main variants of the problem: CP, OP, bounding constraints, and minsum objective function (see Table 5). Additionally, the IPs from both sections are particularly helpful for solving a combination between FD-MmTSP and DFmTSP, where fewer than m depots are part of the input, but the solution still consists of m paths (see Figure 4).

Table 4. IPs' variables and constraints.

IP	Section	Objective Function	Binary Variables	Constraints
IQP1	Section 3.1	quadratic	$\mathcal{O}(n^2m)$	$\mathcal{O}(n^2)$
ILP1		linear	$\mathcal{O}(n^2m)$	$\mathcal{O}(n^2m)$
IQP2	Section 3.2	quadratic	$\mathcal{O}(n^2)$	$\mathcal{O}(n^2)$
ILP2		linear	$\mathcal{O}(n^2)$	$\mathcal{O}(n^2m)$

Table 5. IPs' features and scope.

IP	Dummy Depots	CP	OP	Minsum	Minmax	L	U	FD-M+DF
IQP1	m	✓		✓	✓	✓	✓	✓
ILP1	m	✓	✓	✓	✓	✓	✓	✓
IQP2	m	✓		✓		✓	✓	✓
ILP2	m	✓	✓	✓		✓	✓	✓

In Section 4, we used off-the-shelf optimization software to perform a series of experiments. From them, we observed that the proposed IPs tend to find near-optimal solutions in reasonable amounts of time. Naturally, there is a limit to the practicality of our proposal.

For instance, tight bounding constraints and relatively large values of m make the problem more challenging, and finding feasible solutions for larger graph instances would require more time. Finally, although our work is limited to mTSP, it might be extended to more challenging logistics or scheduling problems, particularly those related to VRP [121–125].

6. Conclusions

The DFmTSP better models real problems where depots are unknown or unnecessary. To our knowledge, this problem and its main variants have not been sufficiently studied. Consequently, this paper introduces different compact IPs for DFmTSP and its main variants. The IPs from Section 3.1 extend an IP for FD-MmTSP and encompass all the main variants of the problem. The IPs from Section 3.2 are more compact in terms of the number of variables and consider most of the variants of the problem. According to a series of empirical tests, the proposed IPs have desirable theoretical properties and practical performance.

There are some future work options to consider. We are especially interested in continue exploiting the attributes of the IPs presented in Section 3.2 since they have only $\mathcal{O}(n^2)$ binary variables. We will try to adapt these IPs to the minmax objective function. Additionally, we will seek more sophisticated formulations that use fewer than m dummy depots or maybe no dummy depots at all. Lastly, we believe some of the ideas exposed in this work may be useful to design exact algorithms for DFmTSP. Moreover, we would like to explore other optimization techniques, like heuristics, metaheuristics, and approximation algorithms, to find near-optimal solutions for larger instances.

Author Contributions: Conceptualization, J.A.C.-A. and J.G.-D.; data curation, J.A.C.-A. and J.G.-D.; formal analysis, J.A.C.-A., J.G.-D., J.C.P.-S. and C.S.; investigation, J.A.C.-A., J.G.-D., J.C.P.-S. and C.S.; methodology, J.A.C.-A. and J.G.-D.; software, J.A.C.-A. and J.G.-D.; supervision, J.G.-D.; validation, J.G.-D., J.C.P.-S. and C.S.; visualization, J.A.C.-A. and J.G.-D.; writing—original draft, J.A.C.-A. and J.G.-D.; writing—review and editing, J.A.C.-A., J.G.-D., J.C.P.-S. and C.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: An implementation of the proposed integer programs is freely available at https://github.com/alex-cornejo/IPDFmTSP (accessed on 5 July 2023) .

Acknowledgments: We acknowledge Gurobi for providing a free-of-charge license for the Gurobi software, which was used to implement the IPs presented in this paper. We also acknowledge Consejo Nacional de Humanidades, Ciencias y Tecnologías (CONAHCYT), Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE), Instituto Tecnológico Superior de Purísima del Rincón, and Centro de Investigación en Matemáticas (CIMAT) for providing the necessary resources for the development of this research.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

CP	Closed paths
CP-DFmTSP	Closed-paths depot-free multiple traveling salesperson problem
CP-FD-MmTSP	Closed-paths fixed-destination multiple-depots multiple traveling salesperson problem
DF	Depot free
DFmTSP	Depot-free multiple traveling salesperson problem
FD-MmTSP	Fixed-destination multiple-depots multiple traveling salesperson problem
ILP	Integer linear program
IP	Integer program
IQP	Integer quadratic program
M	Multiple depots
MmTSP	Multiple-depots multiple traveling salesperson problem

*m*TSP	Multiple traveling salesperson problem
NFD-M*m*TSP	Non-fixed-destination multiple-depots multiple traveling salesperson problem
OP	Open paths
OP-DF*m*TSP	Open-paths depot-free multiple traveling salesperson problem
S	Single-depot
SECs	Subtour elimination constraints
S*m*TSP	Single-depot multiple traveling salesperson problem
TSP	Traveling salesperson problem

Appendix A

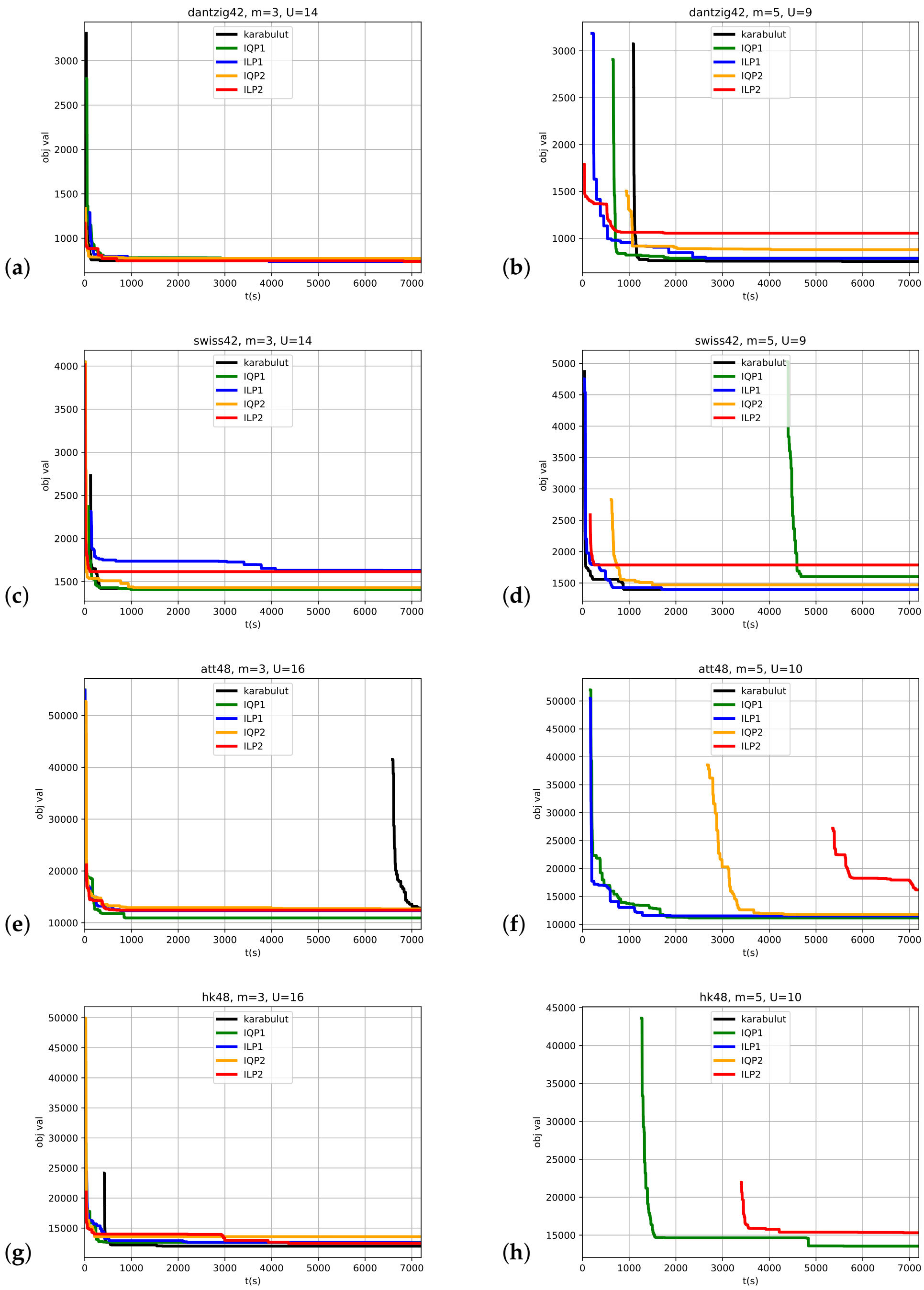


Figure A1. Convergence time reported by Gurobi for the minsum CP-DF*m*TSP with $L = 2$ and tight bounding constraints.

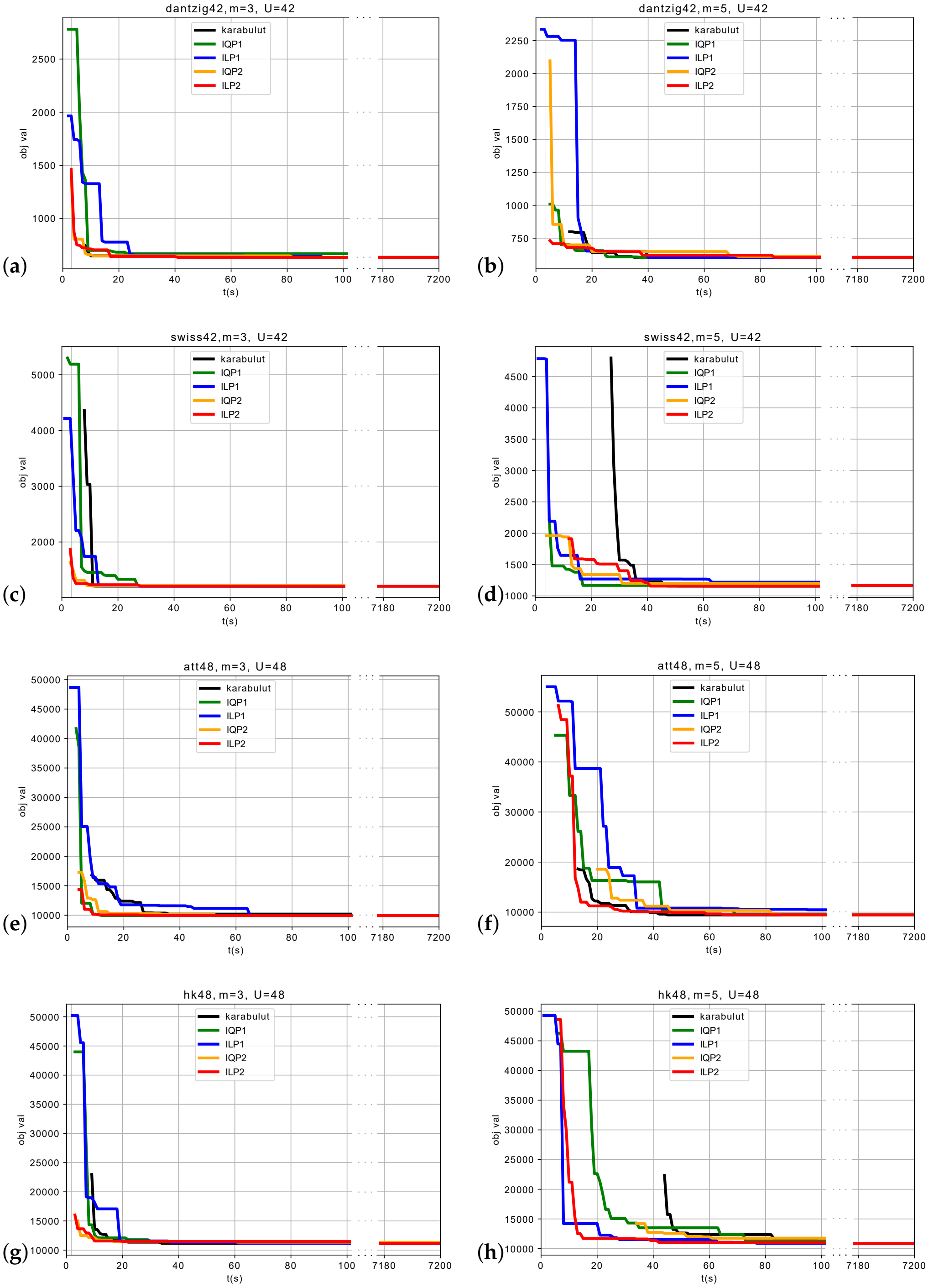


Figure A2. Convergence time reported by Gurobi for the minsum CP-DFmTSP with $L = 2$ and loose bounding constraints.

References

1. Cheikhrouhou, O.; Khoufi, I. A comprehensive survey on the Multiple Traveling salesman Problem: Applications, approaches and taxonomy. *Comput. Sci. Rev.* **2021**, *40*, 100369. [CrossRef]
2. Dantzig, G.B.; Fulkerson, D.R.; Johnson, S.M. *Solution of a Large-Scale Traveling-Salesman Problem*; RAND Corporation: Santa Monica, CA, USA, 1954; pp. 393–410.
3. Dantzig, G.B.; Fulkerson, D.R.; Johnson, S.M. On a linear-programming, combinatorial approach to the traveling-salesman problem. *Oper. Res.* **1959**, *7*, 58–66. [CrossRef]
4. Miller, C.E.; Tucker, A.W.; Zemlin, R.A. Integer programming formulation of traveling salesman problems. *J. ACM* **1960**, *7*, 326–329. [CrossRef]

5. Svestka, J.A.; Huckfeldt, V.E. Computational Experience with an M-salesperson Traveling salesman Algorithm. *Manag. Sci.* **1973**, *19*, 790–799. [CrossRef]
6. Bektas, T. The multiple traveling salesman problem: An overview of formulations and solution procedures. *Omega* **2006**, *34*, 209–219. [CrossRef]
7. Kara, I.; Bektas, T. Integer linear programming formulations of multiple salesman problems and its variations. *Eur. J. Oper. Res.* **2006**, *174*, 1449–1458. [CrossRef]
8. Dantzig, G.B.; Ramser, J.H. The Truck Dispatching Problem. *Manag. Sci.* **1959**, *6*, 80–91. [CrossRef]
9. Mor, A.; Speranza, M.G. Vehicle routing problems over time: A survey. *4OR-Q J. Oper. Res.* **2020**, *18*, 129–149. [CrossRef]
10. Bellmore, M.; Hong, S. Transformation of multisalesperson problem to the standard traveling salesman problem. *J. ACM* **1974**, *21*, 500–504. [CrossRef]
11. Gavish, B. A note on the formulation of the m-salesman traveling salesperson problem. *Manag. Sci.* **1976**, *22*, 704–705. [CrossRef]
12. Svestka, J.A. Response to A note on the formulation of the m-salesman traveling salesperson problem. *Manag. Sci.* **1976**, *22*, 706. [CrossRef]
13. Hong, S.; Padberg, M.W. A note on the symmetric multiple traveling salesman problem with fixed charges. *Oper. Res.* **1977**, *25*, 871–874. [CrossRef]
14. Gavish, B.; Graves, S.C. *The Travelling Salesman Problem and Related Problems*; Massachusetts Institute of Technology, Operations Research Center: Cambridge, MA, USA, 1978.
15. Rao, M.R. A note on the multiple traveling salesmen problem. *Oper. Res.* **1980**, *28*, 628–632. [CrossRef]
16. Discenza, J.H. A more compact formulation of the symmetric multiple traveling salesman problem with fixed charges. *Networks* **1981**, *11*, 73–75. [CrossRef]
17. Jonker, R.; Volgenant, T. An improved transformation of the symmetric multiple traveling salesman problem. *Oper. Res.* **1988**, *36*, 163–167. [CrossRef]
18. Guoxing, Y. Transformation of multidepot multisalesmen problem to the standard travelling salesman problem. *Eur. J. Oper. Res.* **1995**, *81*, 557–560. [CrossRef]
19. Wacholder, E.; Han, J.; Mann, R.C. A neural network algorithm for the multiple traveling salesmen problem. *Biol. Cybern.* **1989**, *61*, 11–19. [CrossRef]
20. Laporte, G.; Yves, N. A cutting planes algorithm for the m-salesmen problem. *J. Oper. Res. Soc.* **1980**, *31*, 1017–1023. [CrossRef]
21. Ali, A.I.; Kennington, J.L. The asymmetric M-travelling salesmen problem: A duality based branch-and-bound algorithm. *Discret. Appl. Math.* **1986**, *13*, 259–276.
22. Ali, A.; Kennington, J. *The M-Travelling Salesmen Problem*; Department of Operations Research, Southern Methodist University: Dallas, TX, USA, 1980.
23. Gavish, B.; Srikanth, K. An optimal solution method for large-scale multiple traveling salesmen problems. *Oper. Res.* **1986**, *34*, 698–717. [CrossRef]
24. Potvin, J.Y.; Lapalme, G.; Rousseau, J.M. A generalized k-opt exchange procedure for the MTSP. *INFOR Inf. Syst. Oper. Res.* **1989**, *27*, 474–481. [CrossRef]
25. Hsu, C.Y.; Tsai, M.H.; Chen, W.M. A study of feature-mapped approach to the multiple travelling salesmen problem. In Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS), Singapore, 11–14 June 1991; pp. 1589–1592.
26. Gromicho, J.; Paixão, J.; Bronco, I. Exact solution of multiple traveling salesman problems. In *Combinatorial Optimization*; Akgül, M., Hamacher, H.W., Tüfekçi, S., Eds.; NATO ASI Series; Springer: Berlin/Heidelberg, Germany, 1992; Volume 82, pp. 291–292, ISBN 978-3-642-77489-8.
27. Russell, R.A. An Effective Heuristic for the M-Tour Traveling salesman Problem with Some Side Conditions. *Oper. Res.* **1977**, *25*, 517–524. [CrossRef]
28. França, P.M.; Gendreau, M.; Laporte, G.; Müller, F.M. The m-traveling salesman problem with minmax objective. *Transp. Sci.* **1995**, *29*, 267–275. [CrossRef]
29. Modares, A.; Somhom, S.; Enkawa, T. A self-organizing neural network approach for multiple traveling salesman and vehicle routing problems. *Int. Trans. Oper. Res.* **1999**, *6*, 591–606. [CrossRef]
30. Somhom, S.; Modares, A.; Enkawa, T. Competition-based neural network for the multiple travelling salesmen problem with minmax objective. *Comput. Oper. Res.* **1999**, *26*, 395–407. [CrossRef]
31. Faigl, J.; Kulich, M.; Preucil, L. Multiple traveling salesmen problem with hierarchy of cities in inspection task with limited visibility. In Proceedings of the 5th Workshop on Self-Organizing Maps, Paris, France, 5–8 September 2005; pp. 91–98.
32. Tang, L.; Liu, J.; Rong, A.; Yang, Z. A multiple traveling salesman problem model for hot rolling scheduling in Shanghai Baoshan Iron & Steel Complex. *Eur. J. Oper. Res.* **2000**, *124*, 267–282.
33. Sofge, D.; Schultz, A.; De Jong, K. Evolutionary Computational Approaches to Solving the Multiple Traveling salesman Problem Using a Neighborhood Attractor Schema. In *Applications of Evolutionary Computing*; Cagnoni, S., Gottlieb, J., Hart, E., Middendorf, M., Raidl, G.R., Eds.; EvoWorkshops, Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2002; Volume 2279, pp. 153–162, ISBN 978-3-540-43432-0.
34. Carter, A.E. Design and Application of Genetic Algorithms for the Multiple Traveling Salesperson Assignment Problem. Ph.D. Thesis, Virginia Polytechnic Institute and State University, Blacksburg, VA, USA, 21 April 2003.

35. Song, C.H.; Lee, K.; Lee, W.D. Extended simulated annealing for augmented TSP and multi-salesmen TSP. In Proceedings of the International Joint Conference on Neural Networks, Portland, OR, USA, 20–24 July 2003; pp. 2340–2343.
36. Bugera, V. Properties of No-Depot MIN-MAX 2-Traveling-salesmen Problem. In *Recent Developments in Cooperative Control and Optimization*; Murphey, S., Pardalos, R., Cagnoni, P.M., Eds.; Cooperative Systems; Springer: Boston, MA, USA, 2004; Volume 3, pp. 45–59, ISBN 978-1-4613-7947-8.
37. Byungsoo, N. Heuristic Approaches for No-Depot k-Traveling Salesmen Problem with a Minmax Objective. Master's Thesis, Texas A&M University, College Station, TX, USA, May 2006.
38. Hou, M.; Liu, D. A novel method for solving the multiple traveling salesmen problem with multiple depots. *Chin. Sci. Bull.* **2012**, *57*, 1886–1892. [CrossRef]
39. Wang, X.; Liu, D.; Hou, M. A novel method for multiple depot and open paths, Multiple Traveling salesmen Problem. In Proceedings of the IEEE 11th International Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl'any, Slovakia, 31 January–2 February 2013.
40. Vandermeulen, I.; Groß, R.; Kolling, A. Balanced Task Allocation by Partitioning the Multiple Traveling Salesperson Problem. In Proceedings of the 18th International Conference on Autonomous Agents and MultiAgent Systems, Montréal, QC, Canada, 13–17 May 2019.
41. Zheng, J.; Hong, Y.; Xu, W.; Li, W.; Chen, Y. An effective iterated two-stage heuristic algorithm for the multiple Traveling Salesmen Problem. *Comput. Oper. Res.* **2022**, *143*, 105772. [CrossRef]
42. Sundar, K.; Rathinamm, S. An exact algorithm for a heterogeneous, multiple depot, multiple traveling salesman problem. In Proceedings of the International Conference on Unmanned Aircraft Systems (ICUAS), Denver, CO, USA, 9–12 July 2015.
43. Yadlapalli, S.; Malik, W.A.; Darbha, S.; Pachter, M. A Lagrangian-based algorithm for a Multiple Depot, Multiple Traveling salesmen Problem. *Nonlinear Anal. Real World Appl.* **2009**, *10*, 1990–1999. [CrossRef]
44. Bektas, T. Formulations and Benders decomposition algorithms for multidepot salesmen problems with load balancing. *Eur. J. Oper. Res.* **2012**, *216*, 83–93. [CrossRef]
45. Aguayo, M.M.; Sarin, S.C.; Sherali, H.D. Solving the Single and Multiple Asymmetric Traveling Salesmen Problems by Generating Subtour Elimination Constraints from Integer Solutions. *IISE Trans.* **2018**, *50*, 45–53. [CrossRef]
46. Campuzano, G.; Obreque, C.; Aguayo, M.M. Accelerating the Miller–Tucker–Zemlin model for the asymmetric traveling salesman problem. *Expert Syst. Appl.* **2020**, *148*, 113229. [CrossRef]
47. Benavent, E.; Martínez, A. Multi-depot Multiple TSP: A polyhedral study and computational results. *Ann. Oper. Res.* **2013**, *207*, 7–25. [CrossRef]
48. Oberlin, P.; Rathinam, S.; Darbha, S. A transformation for a Multiple Depot, Multiple Traveling Salesperson Problem. In Proceedings of the American Control Conference, St. Louis, MO, USA, 10–12 June 2009.
49. Sarin, S.C.; Sherali, H.D.; Judd, J.D.; Tsai, P.F.J. Multiple asymmetric traveling salesmen problem with and without precedence constraints: Performance comparison of alternative formulations. *Comput. Oper. Res.* **2014**, *51*, 64–89. [CrossRef]
50. Assaf, M.; Ndiaye, M. A transformation for multiple depot multiple traveling salesman problem. In Proceedings of the International Conference on Engineering & MIS (ICEMIS), Monastir, Tunisia, 8–10 May 2017.
51. Assaf, M. Transformations for Variants of the Travelling Salesman Problem and Applications. Master's Thesis, American University of Sharjah, Sharjah, United Arab Emirates, April 2017.
52. Kitjacharoenchai, P.; Ventresca, M.; Moshref-Javadi, M.; Lee, S.; Tanchoco, J.M.A.; Brunese, P.A. Multiple traveling salesman problem with drones: Mathematical model and heuristic approach. *Comput. Ind. Eng.* **2019**, *129*, 14–30. [CrossRef]
53. Aguayo, M.M.; Avilés, F.N.; Sarin, S.C.; Sherali, H.D. A Two-Index Formulation for the Fixed-Destination Multi-Depot Asymmetric Travelling Salesman Problem and Some Extensions. *Informatica* **2022**, *33*, 671–692. [CrossRef]
54. Kivelevitch, E.H.; Cohen, K.; Kumar, M. A binary programming solution to the min-max multiple-depots, multiple traveling salesman problem. In Proceedings of the AIAA Infotech@Aerospace (I@A) Conference, Boston, MA, USA, 19–22 August 2013.
55. Qu, H.; Yu, Z.; Tang, H. A columnar competitive model for solving multi-traveling salesman problem. *Chaos Solitons Fractals* **2007**, *31*, 1009–1019. [CrossRef]
56. Matsuura, T.; Numata, K. Solving Min-Max Multiple Traveling salesman Problems by Chaotic Neural Network. In Proceedings of the International Symposium on Nonlinear Theory and Its Applications, Luzern, Switzerland, 14–18 September 2014.
57. Carter, A.E.; Ragsdale, C.T. A new approach to solving the multiple traveling salesperson problem using genetic algorithms. *Eur. J. Oper. Res.* **2006**, *175*, 246–257. [CrossRef]
58. Brown, E.C.; Ragsdale, C.T.; Carter, A.E. A grouping genetic algorithm for the multiple traveling salesperson problem. *Int. J. Inf. Technol. Decis. Mak.* **2007**, *6*, 333–347. [CrossRef]
59. Sze, S.N.; Tiong, W.K. A Comparison between Heuristic and Meta-Heuristic Methods for Solving the Multiple Traveling salesman Problem. *Int. J. Math. Comput. Sci.* **2007**, *1*, 13–16.
60. Zhao, F.; Dong, J.; Li, S.; Yang, X. An improved genetic algorithm for the multiple traveling salesman problem. In Proceedings of the Chinese Control and Decision Conference, Yantai, China, 2–4 July 2008.
61. Singh, A.; Baghel, A.S. A new grouping genetic algorithm approach to the multiple traveling salesperson problem. *Soft Comput.* **2009**, *13*, 95–101. [CrossRef]
62. Zhou, W.; Li, Y. An improved genetic algorithm for multiple traveling salesman problem. In Proceedings of the 2nd International Asia Conference on Informatics in Control, Automation and Robotics, Wuhan, China, 6–7 March 2010.

63. Chen, S.H.; Chen, M.C. Operators of the Two-Part Encoding Genetic Algorithm in Solving the Multiple Traveling salesmen Problem. In Proceedings of the International Conference on Technologies and Applications of Artificial Intelligence, Chung Li, Taiwan, 11–13 November 2011.
64. Király, A.; Abonyi, J. Optimization of Multiple Traveling salesmen Problem by a Novel Representation Based Genetic Algorithm. In *Intelligent Computational Optimization in Engineering. Studies in Computational Intelligence*; Köppen, M., Schaefer, G., Abraham, A., Eds.; Springer: Berlin/Heidelberg, Germany, 2011; Volume 366, pp. 241–269, ISBN 978-3-642-21704-3.
65. Sedighpour, M.; Yousefikhoshbakht, M.; Darani, N.M. An Effective Genetic Algorithm for Solving the Multiple Traveling salesman Problem. *J. Optim. Ind. Eng.* **2011**, *8*, 73–79.
66. Yu, Q.; Wang, D.; Lin, D.; Li, Y.; Wu, C. A Novel Two-Level Hybrid Algorithm for Multiple Traveling Salesman Problems. In *Advances in Swarm Intelligence. ICSI 2012. Lecture Notes in Computer Science*; Tan, Y., Shi, Y., Ji, Z., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7331, pp. 497–503, ISBN 978-3-642-30975-5.
67. Yuan, S.; Skinner, B.; Huang, S.; Liu, L. A new crossover approach for solving the multiple travelling salesmen problem using genetic algorithms. *Eur. J. Oper. Res.* **2013**, *228*, 72–82. [CrossRef]
68. Hosseinabadi, A.A.R.; Kardgar, M.; Shojafar, M.; Shamshirband, S.; Abraham, A. GELS-GA: Hybrid metaheuristic algorithm for solving Multiple Travelling salesman Problem. In Proceedings of the 14th International Conference on Intelligent Systems Design and Applications, Okinawa, Japan, 28–30 November 2014.
69. Alves, R.M.F.; Lopes, C.R. Using genetic algorithms to minimize the distance and balance the routes for the multiple Traveling salesman Problem. In Proceedings of the IEEE Congress on Evolutionary Computation, Sendai, Japan, 25–28 May 2015.
70. Bolaños, R.I.; Toro, E.M.; Granada, M. A population-based algorithm for the multi travelling salesman problem. *Int. J. Ind. Eng. Comput.* **2016**, *7*, 245–256. [CrossRef]
71. Lu, Z.; Zhang, K.; He, J.; Niu, Y. Applying K-means Clustering and Genetic Algorithm for Solving MTSP. In Proceedings of the 11th International Conference, BIC-TA, Xi'an, China, 28–30 October 2016.
72. Lo, K.M.; Yi, W.Y.; Wong, P.K.; Leung, K.S.; Leung, Y.; Mak, S.T. A Genetic Algorithm with New Local Operators for Multiple Traveling salesman Problems. *Int. J. Comput. Intell. Syst.* **2018**, *11*, 692–705. [CrossRef]
73. Singh, D.R.; Singh, M.K.; Singh, T.; Prasad, R. Genetic Algorithm for Solving Multiple Traveling Salesmen Problem using a New Crossover and Population Generation. *Comput. Sist.* **2018**, *22*, 491–503. [CrossRef]
74. Xu, X.; Yuan, H.; Liptrott, M.; Trovati, M. Two phase heuristic algorithm for the multiple-travelling salesman problem. *Soft Comput.* **2018**, *22*, 6567–6581. [CrossRef]
75. Zhou, H.; Song, M.; Pedrycz, W. A comparative study of improved GA and PSO in solving multiple traveling salesmen problem. *Appl. Soft Comput.* **2018**, *64*, 564–580. [CrossRef]
76. Harrath, Y.; Salman, A.F.; Alqaddoumi, A.; Hasan, H.; Radhi, A. A novel hybrid approach for solving the multiple traveling salesmen problem. *Arab. J. Basic Appl. Sci.* **2019**, *26*, 103–112. [CrossRef]
77. Jiang, C.; Wan, Z.; Peng, Z. A new efficient hybrid algorithm for large scale multiple traveling salesman problems. *Expert Syst. Appl.* **2020**, *139*, 112867. [CrossRef]
78. Al-Furhud, M.A.; Ahmed, Z.H. Experimental Study of a Hybrid Genetic Algorithm for the Multiple Travelling salesman Problem. *Math. Probl. Eng.* **2020**, *2020*, 3431420. [CrossRef]
79. Wang, Z.; Fang, X.; Li, H.; Jin, H. An Improved Partheno-Genetic Algorithm With Reproduction Mechanism for the Multiple Traveling Salesperson Problem. *IEEE Access* **2020**, *8*, 102607–102615. [CrossRef]
80. Singamsetty, P.; Thenepalle, J.K. An efficient genetic algorithm for solving open multiple travelling salesman problem with load balancing constraint. *Decis. Sci. Lett.* **2021**, *10*, 525–534. [CrossRef]
81. Al-Taani, A.T.; Al-Alfifi, L.M. Solving the Multiple Traveling Salesman Problem Using Memetic Algorithm. *Artif. Intell. Evol.* **2022**, *3*, 27–40. [CrossRef]
82. Lou, P.; Xu, K.; Yan, J.; Xiao, Z. An Improved Partheno-Genetic Algorithm for Open Path Multi-Depot Multiple Traveling Salesmen Problem. *J. Phys. Conf. Ser.* **2021**, *1848*, 012002. [CrossRef]
83. Chandran, N.; Narendran, T.T.; Ganesh, K. A clustering approach to solve the multiple travelling salesmen problem. *Int. J. Ind. Syst. Eng.* **2006**, *1*, 372–387. [CrossRef]
84. Junjie, P.; Dingwei, W.; Preucil, L. An Ant Colony Optimization Algorithm for Multiple Travelling salesman Problem. In Proceedings of the First International Conference on Innovative Computing, Beijing, China, 30 August–1 September 2006.
85. Liu, W.; Li, S.; Zhao, F.; Zheng, A. An ant colony optimization algorithm for the Multiple Traveling Salesmen Problem. In Proceedings of the 4th IEEE Conference on Industrial Electronics and Applications, Xi'an, China, 25–27 May 2009.
86. Ghafurian, S.; Javadian, N. An ant colony algorithm for solving fixed destination multi-depot multiple traveling salesmen problems. *Appl. Soft Comput.* **2011**, *11*, 1256–1262. [CrossRef]
87. Yousefikhoshbakht, M.; Sedighpour, M. A combination of sweep algorithm and elite ant colony optimization for solving the multiple traveling salesman problem. *Proc. Rom. Acad. -Ser. Math. Physics, Tech. Sci. Inf. Sci.* **2012**, *13*, 295–301.
88. Lu, L.C.; Yue, T.W. Mission-oriented ant-team ACO for min–max MTSP. *Appl. Soft Comput.* **2019**, *76*, 436–444. [CrossRef]
89. Farisi, O.I.R.; Setiyono, B.; Danandjojo, R.I. A hybrid approach to multi-depot multiple traveling salesman problem based on firefly algorithm and ant colony optimization. *Int. J. Artif. Intell. (IJ-AI)* **2021**, *10*, 910–918. [CrossRef]
90. Costa-Salas, Y.J.; Abreu-Ledón, R.; Coello-Machado, J.I.; Nowé, A. Multi-type ant colony system for solving the multiple traveling salesman problem. *Rev. TéCnica Fac. Ing. Univ. Del Zulia* **2012**, *35*, 311–320.

91. Necula, R.; Breaban, M.; Raschip, M. Performance Evaluation of Ant Colony Systems for the Single-Depot Multiple Traveling Salesman Problem. In *Hybrid Artificial Intelligent Systems*; Onieva, E., Santos, I., Osaba, E., Quintián, H., Corchado, E., Eds.; Springer: Cham, Switzerland, 2015; Volume 9121, pp. 257–268, ISBN 978-3-319-19643-5.
92. Kivelevitch, E.; Cohen, K.; Kumar, M. A Market-based Solution to the Multiple Traveling salesmen Problem. *J. Intell. Robot. Syst.* **2013**, *72*, 21–40. [CrossRef]
93. Scott, D.; Manyam, S.G.; Casbeer, D.W.; Kumar, M. Market Approach to Length Constrained Min-Max Multiple Depot Multiple Traveling Salesman Problem. In Proceedings of the 2020 American Control Conference, Denver, CO, USA, 1–3 July 2020.
94. Larki, H.; Yousefikhoshbakht, M. Solving the Multiple Traveling salesman Problem by a Novel Meta-heuristic Algorithm. *J. Optim. Ind. Eng.* **2014**, *16*, 55–63.
95. Rostami, A.S.; Mohanna, F.; Keshavarz, H.; Hosseinabadi, A.A.R. Solving Multiple Traveling Salesman Problem using the Gravitational Emulation Local Search Algorithm. *Appl. Math. Inf. Sci.* **2015**, *2*, 699–709.
96. Soylu, B. A general variable neighborhood search heuristic for multiple traveling salesmen problem. *Comput. Ind. Eng.* **2015**, *90*, 390–401. [CrossRef]
97. Wang, Y.; Chen, Y.; Lin, Y. Memetic algorithm based on sequential variable neighborhood descent for the minmax multiple traveling salesman problem. *Comput. Ind. Eng.* **2017**, *106*, 105–122. [CrossRef]
98. Dong, X.; Zhang, H.; Xu, M.; Shen, F. Hybrid genetic algorithm with variable neighborhood search for multi-scale multiple bottleneck traveling salesmen problem. *Future Gener. Comput. Syst.* **2021**, *114*, 229–242. [CrossRef]
99. Wang, M.; Xin, B.; Wang, Q. A general variable neighborhood search for the multiple depots multiple traveling salesmen problem. In Proceedings of the The International Workshop on Advanced Computational Intelligence and Intelligent Informatics (IWACIII2021), Beijing, China, 31 October–3 November 2021.
100. Venkatesh, P.; Singh, A. Two metaheuristic approaches for the multiple traveling salesperson problem. *Appl. Soft Comput.* **2015**, *26*, 74–89. [CrossRef]
101. Chen, Y.; Jia, Z.; Ai, X.; Yang, D.; Yu, J. A modified two-part wolf pack search algorithm for the multiple traveling salesmen problem. *Appl. Soft Comput.* **2017**, *61*, 714–725. [CrossRef]
102. Alazzam, H.; Alsmady, A.; Mardini, W. Solving Multiple Traveling salesmen Problem using Discrete Pigeon Inspired Optimizer. In Proceedings of the 11th International Conference on Information and Communication Systems (ICICS), Irbid, Jordan, 7–9 April 2020.
103. Hu, Y.; Yao, Y.; Lee, W.S. A reinforcement learning approach for optimizing multiple traveling salesman problems over graphs. *Knowl.-Based Syst.* **2020**, *204*, 106244. [CrossRef]
104. Karabulut, K.; Öztop, H.; Kandiller, L.; Tasgetiren, M.F. Modeling and optimization of multiple traveling salesmen problems: An evolution strategy approach. *Comput. Oper. Res.* **2021**, *129*, 105192. [CrossRef]
105. He, P.; Hao, J.-K. Memetic search for the minmax multiple traveling salesman problem with single and multiple depots. *Eur. J. Oper. Res.* **2023**, *307*, 1055-1070. [CrossRef]
106. Zhang, Y.; Han, X.; Dong, Y.; Xie, J.; Xie, G.; Xu, X. A novel state transition simulated annealing algorithm for the multiple traveling salesmen problem. *J. Supercomput.* **2021**, *77*, 11827–11852. [CrossRef]
107. Hamza, A.; Darwish, A.H.; Rihawi, O. A new local search for the bees algorithm to optimize multiple traveling salesman problem. *Intell. Syst. Appl.* **2023**, *18*, 200242. [CrossRef]
108. Averbakh, I.; Berman, O. A heuristic with worst-case analysis for minimax routing of two travelling salesmen on a tree. *Discret. Appl. Math.* **1996**, *68*, 17–32. [CrossRef]
109. Averbakh, I.; Berman, O. (p-1)(p+1)-approximate algorithms for p-traveling salesmen problems on a tree with minmax objective. *Discret. Appl. Math.* **1997**, *75*, 201–216. [CrossRef]
110. Nagamoshi, H.; Okada, K. A faster 2-approximation algorithm for the minmax p-traveling salesmen problem on a tree. *Discret. Appl. Math.* **2004**, *140*, 103–114. [CrossRef]
111. Rathinam, S.; Sengupta, R. *Lower and Upper Bounds for a Symmetric Multiple Depots, Multiple Travelling salesman Problem*; Research Report, UCB-ITS-RR-2006-2; Institute of Transportation Studies, University of California at Berkeley: Berkeley, CA, USA, 2006,
112. Xu, Z.; Rodrigues, B. A 3/2-Approximation Algorithm for Multiple Depot Multiple Traveling salesman Problem. In *Lecture Notes in Computer Science*; Kaplan, H., Ed.; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6139, pp. 127–138, ISBN 978-3-642-13730-3.
113. Xu, Z.; Xu, L.; Rodrigues, B. An analysis of the extended Christofides heuristic for the k-depot TSP. *Oper. Res. Lett.* **2011**, *39*, 218–223. [CrossRef]
114. Xu, Z.; Rodrigues, B. An extension of the Christofides heuristic for the generalized multiple depot multiple traveling salesmen problem. *Eur. J. Oper. Res.* **2017**, *257*, 735–745. [CrossRef]
115. Xu, L.; Xu, Z.; Xu, D. Exact and approximation algorithms for the min–max *k*-traveling salesmen problem on a tree. *Eur. J. Oper. Res.* **2013**, *227*, 284–292. [CrossRef]
116. Pérez-Escalona, P.; Rapaport, I.; Soto, J.; Vidal, I. The Multiple Traveling Salesman Problem on Spiders. In Proceedings of the 47th International Conference on Current Trends in Theory and Practice of Computer Science, SOFSEM, Bolzano-Bozen, Italy, 25–29 January 2021.
117. He, P.; Hao, J.-K. Hybrid search with neighborhood reduction for the multiple traveling salesman problem. *Comput. Oper. Res.* **2022**, *142*, 105726. [CrossRef]

118. Mantha, B.R.K.; Jung, M.K.; García de Soto, B.; Menassa, C.C.; Kamat, V.R. Generalized task allocation and route planning for robots with multiple depots in indoor building environments. *Autom. Constr.* **2020**, *119*, 103359. [CrossRef]
119. Gurobi Optimization, LLC Gurobi Optimizer Reference Manual. 2022. Available online: https://www.gurobi.com (accessed on 20 June 2023).
120. Reinelt, G. TSPLIB—A traveling salesman problem library. *ORSA J. Comput.* **2021**, *3*, 376–384. [CrossRef]
121. Ransikarbum, K.; Wattanasaeng, N.; Madathil, S.C. Analysis of multi-objective vehicle routing problem with flexible time windows: The implication for open innovation dynamics. *J. Open Innov. Technol. Mark. Complex.* **2023**, *9*, 100024. [CrossRef]
122. Zhou, H.; Qin, H.; Cheng, C.; Rousseau, L.-M. An exact algorithm for the two-echelon vehicle routing problem with drones. *Transp. Res. Part B Methodol.* **2023**, *168*, 124–150. [CrossRef]
123. Sar, K.; Ghadimi, P. A systematic literature review of the vehicle routing problem in reverse logistics operations. *Comput. Ind. Eng.* **2023**, *177*, 109011. [CrossRef]
124. Moghdani, R.; Salimifard, K.; Demir, E.; Benyettou, A. The green vehicle routing problem: A systematic literature review. *J. Clean. Prod.* **2023**, *279*, 123691. [CrossRef]
125. Menchaca-Méndez, A.; Montero, E.; Flores-Garrido, M.; Miguel-Antonio, L. An algorithm to compute time-balanced clusters for the delivery logistics problem. *Eng. Appl. Artif. Intell.* **2023**, *111*, 104795. [CrossRef]