algorithms

MDPI

Article

Exact and Approximate Linear Models for the Multi-Vehicle Probabilistic Covering Tour Problem

Roberto Montemanni [1,*] and Derek H. Smith [2]

[1] Department of Sciences and Methods for Engineering, University of Modena and Reggio Emilia, 42122 Reggio Emilia, Italy
[2] Faculty of Computing, Engineering and Science, University of South Wales, Pontypridd CF37 1DL, UK; derek.smith@southwales.ac.uk
* Correspondence: roberto.montemanni@unimore.it; Tel.: +39-0522-522-126

Abstract

The Multi-Vehicle Probabilistic Covering Tour Problem aims at designing the tours of a fleet of vehicles characterized by a limited range that has to distribute goods to some distribution facilities, in order for those goods to subsequently reach the final customers. Each truck distributes directly only to the facilities, and the delivered goods reach the final customers with a given probability. The aim of the problem is to maximize the expected quantity of goods delivered to the final customers. The problem can be modeled as a mixed integer program with a non-linear objective function, as already presented in the literature together with a solving method based on a linear model with an exponential number of constraints. The contributions of the present work are a new model and two new linear approximation models able to provide tight lower and upper bounds for the optimal solution of the original problem. An extensive computational campaign on the instances available in the literature validates the new approaches we propose, providing several improved heuristic solutions and upper bounds with respect to the previous state-of-the-art. Several instances are also closed here for the first time.

Keywords: covering tour problem; multi-vehicle; probabilistic settings; exact models; approximation models

1. Introduction

The Multi-Vehicle Probabilistic Covering Tour Problem (MVPCTP), originally proposed in [1], is the subject of the present study. This problem concerns the planning of tours for a fleet of vehicles distributing goods to customers, each vehicle having a limited range. Each truck leaves the depot, visits some given facilities from which customers are subsequently served, and goes back to the depot, without exceeding the given range limit. Once a truck visits a facility, the goods are delivered to each of the customers with a given probability. The framework models situations in which there is uncertainty about the success of delivering to the customers from the facilities (e.g., disaster management) and in which the capacity of the trucks is a limited resource. The model assumes that not all the customers will receive their goods, and it aims at maximizing the probability of delivery to all the customers. What happens to customers that are not serviced is beyond the scope of the approach. The MVPCTP is a generalization of the Covering Tour Problem (CTP), originally introduced in [2]. In this problem, it is assumed that when a customer is within a given distance of a facility that is visited then the goods automatically reach the

Academic Editor: Pengyu Yan

Received: 23 June 2026
Revised: 30 July 2026
Accepted: 1 August 2026
Published: 3 August 2026

customers, instead of probabilistically as for the MVPCTP. The authors of [1] proposed a non-linear integer programming model and solving methods based on branch and bound and metaheuristic frameworks.

Applications of the MVPCTP can be found in the tasks of remote sensing drones for patrolling or searching applications. Typical domains are in agriculture, surveillance and disaster management [3,4]. The interested reader is referred to survey [5] for a more comprehensive review.

The MVPCTP is one of the set of problems that generalize the classic Vehicle Routing Problem (VRP), where a fleet of vehicles has to visit a set of locations minimizing either the total distance traveled or the number of vehicles employed. We refer the interested reader to book [6] for a complete overview of classic VRP generalizations. Stochastic variants of the VRP have also been treated extensively in the literature. The interested reader can find more general information in [7].

A brief survey of the literature related to the MVPCTP follows. The CTP was originally proposed in [2], where the problem was attacked by branch-and-cut and constructive heuristics. Later, ref. [8] proposed an adaptive large neighborhood search metaheuristic based on giant tours to solve the problem heuristically. A bi-objective version of the problem, optimizing both the length of the tour and the cover provided, was introduced in [9]. The authors developed a multi-objective evolutionary algorithm and a branch-and-cut algorithm. A real application about routing of healthcare teams motivated the study presented in [10], where the multi-vehicle Covering Tour Problem (m-CTP) was formalized. The authors proposed a mathematical formulation together with a set of heuristic algorithms. Subsequently, ref. [11] devised a successful branch-and-price approach for the problem. In paper [12] a different branch-and-price algorithm for the m-CTP was presented and validated by a vast computational campaign. A new version of the problem, where route-balancing is considered, was introduced in [13]. A new flow-based formulation together with a branch-and-cut approach and biased random-key genetic heuristics were discussed by the authors. A further variation of the m-CTP, where coverage requirements and speed optimization were added, was formulated and solved via branch-and-price in [14]. Exact and heuristic algorithms for a variation of the problem where the constraints are on the number of vertices in each route were discussed in [15]. A bi-objective variant of the m-CTP was introduced and addressed via column-generation in [16]. A variation of the m-CTP with multiple depots was introduced in [17]. Two integer programming formulations and a hybrid metaheuristic were proposed and tested. A more elaborate variant of the m-CTP was introduced in [18], the aim being to find a set of vehicle tours that minimize the sum of arrival times at each visited location. The problem was motivated by a real application related to contracts with the customers. The authors proposed a mixed integer linear programming formulation and a heuristic approach based on a greedy randomized adaptive search procedure. The authors of [19] introduced a further variant named Multi-Vehicle Multi-Covering Tour Problem (mm-CTP), in which the customers must be covered several times in the planning horizon. A formulation, a branch-and-cut solving procedure and a genetic algorithm were the contributions of the authors. Two other metaheuristic approaches for a variant of the same problem, where the number of customers in each tour is constrained, were discussed in [20]. More recently, ref. [21] proposed a branch-cut-and-price approach incorporating some new valid inequalities for the m-CTP. A variation of the m-CTP designed to model waste collection was treated in [22], where several exact and heuristic methods were proposed. A study on the Covering Tour Location Routing Problem, integrating covering, routing and facility location decisions into a single model, was discussed in [23]. A summary of the contributions appeared in the literature for the CTP, and some of its variants can be found in Table 1. Finally, model-based approaches to attack probabilistic optimization problems by solving a sequence of deterministic models have been presented in [24,25].

Table 1. Summary of the main contributions in the literature related to the Covering Tour Problem and its variants.

Reference	Problem	Solution Approach	Main Contribution
Gendreau et al. [2]	CTP	Branch-and-cut; constructive heuristics	Original definition of the Covering Tour Problem and first exact and heuristic methods.
Jozefowiez et al. [9]	Bi-objective CTP	Evolutionary algorithm; branch-and-cut	Simultaneous optimization of routing cost and coverage.
Vargas et al. [8]	CTP	Adaptive Large Neighborhood Search	Effective giant-tour based metaheuristic.
Hachicha et al. [10]	m-CTP	MILP; heuristics	First formulation of the multi-vehicle variant motivated by healthcare applications.
Lopes et al. [11]	m-CTP	Branch-and-price	Exact algorithm based on column generation.
Jozefowiez [12]	m-CTP	Branch-and-price	Alternative branch-and-price algorithm with extensive computational study.
Hà et al. [15]	Cardinality-constrained m-CTP	Exact and heuristic algorithms	Route cardinality constraints.
Allahyari et al. [17]	Multi-depot m-CTP	MILP; hybrid metaheuristic	Extension to multiple depots.
Flores-Garza et al. [18]	Arrival-time m-CTP	MILP; GRASP	Minimization of customer arrival times.
Pham et al. [19]	Multi-Covering Tour Problem	Branch-and-cut; genetic algorithm	Multiple coverage requirements.
Glize et al. [16]	Bi-objective m-CTP	Column generation	Multi-objective optimization.
Kammoun et al. [20]	Constrained mm-CTP	Metaheuristics	Efficient heuristics for constrained routes.
Margolis et al. [14]	Speed-aware m-CTP	Branch-and-price	Joint optimization of routes, speeds and coverage.
Ota et al. [13]	Balanced m-CTP	Flow formulation; branch-and-cut; BRKGA	Route balancing with new flow formulation.
Karaoğlan et al. [1]	MVPCTP	Branch-and-cut; VNS	Introduction of probabilistic customer coverage under vehicle range constraints.
Oliveira et al. [21]	m-CTP	Branch-cut-and-price	New valid inequalities, stronger exact algorithm and new benchmark instances.
Fischer et al. [22]	Capacitated road-network m-CTP	MILP; two-phase heuristic	Capacitated covering tours on road networks motivated by waste collection.
Hagn et al. [23]	Covering Tour Location Routing Problem	MILP; matheuristic	Integration of covering, routing and facility location decisions.

The contribution of the present work is twofold: a new model that modifies the one originally introduced in [1] to represent more effectively the vehicle routing component of the MVPCTP within linearizations. This is achieved by defining a set of variables for each vehicle tour instead of having a common set of variables for all the tours. Note that

https://doi.org/10.3390/a19080643

this increases the number of variables, but it should hopefully give an advantage to the new model in all instances apart from very large ones; two new approximation models, able to provide tight upper bounds and high-quality heuristic solutions for the MVPCTP are then presented. All the new methods were evaluated experimentally in the instances available in the literature, and they were shown to have been instrumental in improving the best-known lower and upper bounds, which in turn led to the proven optimal solution in several instances.

The paper is organized as follows. In Section 2 the MVPCTP is formally defined, and an example is provided. A non-linear mixed integer programming model for the problem (based on that originally disclosed in [1]) is reported in Section 3. Two linear models characterized by an exponential number of inequalities to deal with the linearization of the originally non-linear objective function of the problem are discussed in Section 4. Section 5 presents two new linear models providing approximating solutions for the MVPCTP without the need for any exponentially large set of inequalities. The extensive experimental campaign conducted is summarized in Section 6, where some implementation details are also disclosed, together with a description of the instances adopted. Finally, some conclusions about the potential of the new models are drawn in Section 7.

2. Problem Description

The Multi-Vehicle Probabilistic Covering Tour Problem (MVPCTP), originally introduced in [1], can be formally described as follows. The notation adopted throughout the paper is summarized in Table 2. A directed graph $G = (\{0\} \cup V \cup W, A)$ is used to represent the problem, where the vertices have the following meanings: 0 is the depot, V is the set of facilities and W is the set of customers. A demand $q_j > 0$ is associated with each customer $j \in W$. The arc set A contains all the possible connections between the facilities and between the depot and the facilities. A distance d_{ij} is associated with each arc $(i, j) \in A$. A set K of homogeneous vehicles is given. Each vehicle leaves the depot 0, visits a subset of the facilities V and returns to the depot, traveling no more than a given allowed maximum distance L. Each facility $i \in V$ can be visited at most once by the fleet of vehicles. If a vehicle visits facility $i \in V$ then it serves customer $j \in W$ with a given probability $0 \leq p_{ij} < 1$. The objective is to select a set of feasible tours for the vehicles such that the expected total demand delivered to the customers is maximized according to the probabilities given. A small example of the Multi-Vehicle Probabilistic Covering Tour Problem is provided in Figure 1.

Let y_i take a value of 1 if any of the vehicles visits the facility $i \in V$, 0 otherwise. As pointed out in [1], given a feasible assignment of the y variables the total probability that customer vertex $j \in W$ is served can be expressed as $1 - \prod_{i \in V}(1 - p_{ij})^{y_i}$, where $(1 - p_{ij})^{y_i}$ is used to express the probability that customer j is not visited from facility i. Therefore, the expected total demand delivered, given a feasible assignment represented by y variables, can be expressed as follows:

$$\mathbb{E}(y) = \sum_{j \in W} q_j \left(1 - \prod_{i \in V} (1 - p_{ij})^{y_i} \right) \tag{1}$$

Table 2. Notation used throughout the paper.

Symbol	Description
Sets and indices	
$G = (\{0\} \cup V \cup W, A)$	Directed graph representing the problem.
0	Depot vertex.

Table 2. *Cont.*

Symbol	Description
Sets and indices	
V	Set of candidate facilities.
W	Set of customers.
A	Set of directed arcs.
K	Set of homogeneous vehicles.
S	Subset of facilities used in subtour elimination constraints.
i, j, k, l	Indices of facilities or customers (depending on context).
Parameters	
d_{ij}	Distance associated with arc $(i, j) \in A$.
q_j	Demand of customer $j \in W$.
p_{ij}	Probability that facility i serves customer j.
L	Maximum travel distance allowed for each vehicle.
$\|K\|$	Number of available vehicles.
γ_i	Generic coefficient used in the approximation proofs.
$E(y)$	Expected total demand served by solution $\mathbf{y}$.
Decision variables	
y_i	Binary variable equal to 1 if facility i is visited by any vehicle, 0 otherwise.
x_{ij}	Binary variable equal to 1 if arc (i, j) is traversed.
t_i	Cumulative travel distance when a vehicle reaches facility i.
w_j	Continuous variable representing the linearized contribution of customer j to the objective function.
z_{ij}^k	Binary variable equal to 1 if vehicle k traverses arc (i, j).
r_{ij}	Binary variable equal to 1 if both facilities i and j are visited.
s_{ijk}	Binary variable equal to 1 if facilities i, j, and k are all visited.
Auxiliary notation	
y	Binary vector of facility-selection variables.
y^*	Reference point used to define tangent inequalities in the linearization.
$f(y)$	Nonlinear coverage function used in the objective linearization.

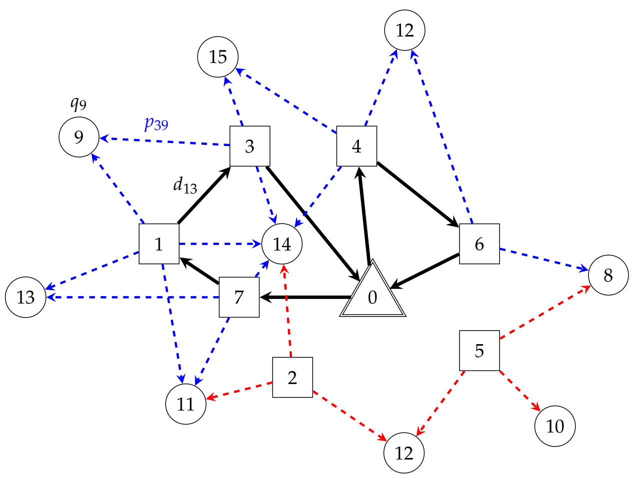

Figure 1. Example of an MVPCTP instance, where the depot is represented as a triangle, the facilities as squares and the customers as circles. The routes of the trucks are represented in black, while the non-null visiting probabilities are depicted in dotted blue for those collected, and in dotted red for those not collected. On the top left of the figure travel distances d, demands q and probability p are also placed. In the rest of the figure these elements are omitted. Note that facilities 2 and 5 are not visited in the solution depicted. This causes the probability of being visited to be zero for customers 10 and 12 and to be reduced for customers 8, 11 and 14 (since they are reached from other facilities).

3. A Non-Linear Model

The MVPCTP can be modeled as a non-linear problem as follows, as presented in [1].

A first set of variables y follows the definition anticipated in Section 2: y_i takes a value of 1 if any of the vehicles visits the facility $i \in V$, 0 otherwise. Another set of variables x is defined, such that x_{ij} takes value of 1 if arc $(i,j) \in A$ is traversed, 0 otherwise. Finally, a new set of variables t is such that t_i contains the cumulative distance traveled by a vehicle when it eventually reaches vertex $i \in V$. The following model, in the spirit of that originally introduced in [1], can be derived:

$$(NL) \quad \max \sum_{j \in W} q_j \left(1 - \prod_{i \in V} (1 - p_{ij})^{y_i} \right) \tag{2}$$

$$s.t. \quad \sum_{k \in V \cup \{0\}, k \neq i} x_{ki} = y_i \qquad i \in V \tag{3}$$

$$\sum_{i \in V} x_{0i} \leq |K| \tag{4}$$

$$\sum_{k \in V \cup \{0\}, k \neq i} (x_{ki} - x_{ik}) = 0 \qquad i \in V \tag{5}$$

$$t_i \geq d_{0i} x_{0i} \qquad i \in V \tag{6}$$

$$t_k \geq t_i - L + (L + d_{ik}) x_{ik} \qquad i,k \in V, k \neq i \tag{7}$$

$$x_{ij} \in \{0,1\} \qquad i,j \in V \cup \{0\} \tag{8}$$

$$y_i \in \{0,1\} \qquad i \in V \cup \{0\} \tag{9}$$

$$t_i \in [0, L - d_{i0}] \qquad i \in V \tag{10}$$

The objective function (2) maximizes the expected demand delivered, according to (1). Equality (3) defines y variables in terms of the values of the x variables. Constraint (4) makes sure that no more than $|K|$ vehicles are deployed. Constraint (5) enforces flow-conservation and, together with constraints (6) and (7), ensures feasible tours for the vehicles. Inequalities (6) and (7) regulate the distance traveled by the vehicles. The remaining constraints (8)–(10) define the domain of the variables. Note that constraint (10) contributes to imposing that the traveling distance budget L is not exceeded by the vehicles.

4. Linear Models

In this section two linear models for the MVPCTP are described. The first was presented in [1], while the second is introduced here for the first time. Both the models are based on the same linearization of the objective function (2), while they differ in their modeling of the *multiple traveling salesman problem* component of the problem. The first one uses 2-indices variables, while the new one uses 3-indices variables.

4.1. Model I2

An effective linearization for the model (NL) was proposed in [1]. Proposing a linearization is effective when formulations are processed by solvers in order to find high-quality solutions, since state-of-the-art linear solvers are able to handle substantially larger problems than state-of-the-art non-linear solvers. The linearization proposed is based on a theoretical result showing that the objective function (2) is concave, and therefore the tangents of the components $f(y_j) = 1 - \prod_{i \in V}(1 - p_{ij})^{y_i}$ (which is equivalent to $1 - \prod_{i \in V}(1 - p_{ij} y_i)) \; \forall j \in W$) can be used to linearize the model NL in a piecewise fashion. We refer the interested reader to [1] for all the details and theoretical results relevant to the linearization.

The new model $I2$ can be described as follows:

$$(I2)\quad \max \sum_{j\in W} q_j w_j \tag{11}$$

$$\begin{aligned} s.t.\ & (3)\text{–}(10) \\ & w_j \le \sum_{k\in V} -\ln(1-p_{kj})\prod_{i\in V}(1-p_{ij})^{y_i^*} y_k + 1 - \prod_{i\in V}(1-p_{ij})^{y_i^*} + \\ & \quad \sum_{k\in V}\ln(1-p_{kj})\prod_{i\in V}(1-p_{ij})^{y_i^*} y_k^* \qquad y^* \in [0,1]^{|V|-1} \end{aligned} \tag{12}$$

$$w_j \ge 0 \qquad j \in W \tag{13}$$

The new constraint (12) binds each variable w_j to take a value less than or equal to that of the tangent of $f(y_j^*)$ for each possible (feasible) assignment y^*, according to the current assignment of variables y within the model. Note that there is an exponential number of constraints. The domain of variables w is defined by inequality (13), while their values are combined, weighted by coefficients q, within the new objective function (11), that exploits the new variables w.

4.2. Model I3

An alternative model using a new set of variables with three indices instead of the x variables with two indices, but not requiring timing variables t and the related constraints, can be derived and is described in the present section. Note that this model is one of the novel contributions of this paper.

On top of the variables y defined in Section 2, let us also define a new set of variables z such that z_{ij}^k takes a value of 1 if arc $(i,j) \in A$ is traversed by the tour of truck $k \in K$. The following model, introduced here for the first time, can be derived:

$$(I3)\quad \max \sum_{j\in W} q_j w_j$$

$$s.t.\ (9), (12), (13)$$

$$\sum_{k\in K}\sum_{l\in V\cup\{0\}, l\ne i} z_{li}^k = y_i \qquad i \in V \tag{14}$$

$$\sum_{l\in V\cup\{0\}, l\ne i} z_{li}^k - \sum_{l\in V\cup\{0\}, l\ne i} z_{il}^k = 0 \qquad k\in K, i \in V\cup\{0\} \tag{15}$$

$$\sum_{i\in S}\sum_{j\in S, j\ne i} z_{ij}^k \le |S|-1 \qquad k \in K, S\subseteq V, |S|\ge 2 \tag{16}$$

$$\sum_{i\in V\cup\{0\}}\sum_{j\in V\cup\{0\}, j\ne i} d_{ij} z_{ij}^k \le L \qquad k\in K \tag{17}$$

$$z_{ij}^k \in \{0,1\} \qquad k\in K, i,j\in V\cup\{0\} \tag{18}$$

The model relies on concepts and constraints introduced in the previous sections, while the novel part is that related to the description of the tours of the vehicles, which are now modeled through separate sets of variables, one for each route. Constraint (14) sets variables y and at the same time imposes that each facility is visited at most once. Constraint (15) imposes that for each tour k and for each vertex i the balance of active incoming and outgoing arcs has to match (being either 0 or 1). Constraint (16) is a classic subtour elimination constraint [26] and enforces that the tours of the vehicles are valid by imposing that for each subset of $|S|$ vertices (not including the depot 0) no more than $|S|-1$ arcs can be active, otherwise a loop would be formed. Note that the combination of constraints (15) and (16) allows only feasible tours. Inequality (17) imposes that the tours of the vehicles do not exceed the allowed maximum distance L. Finally, the domains of the new variables z are defined by constraint (18). Note that model $I3$ is suitable to handle a heterogeneous fleet of vehicles, although this feature is not used in the present study. A side effect is that the model has symmetry issues, since the tours are interchangeable. However, in the experimental part of the study (Section 6), the solver adopted has very strong symmetry-breaking strategies embedded, making this issue less prominent.

5. Approximation Models

It is possible to derive Mixed-Integer Linear Programs able to provide (tight) approximations for the optimal solution of the MVPCTP following a different approach. In detail, we aim to substitute the objective function (2) with linear functions providing an approximation of the original one, without the need for an exponential number of constraints as in the models seen in Section 4, although they are not designed to provide a proven exact solution. As we will show, the models will provide valid upper bounds (estimations by excess) for the values of the optimal solutions of the MVPCTPs, together with heuristic solutions that—once evaluated according to the original objective function of the MVPCTP—provide lower bounds for the values of the optimal solutions of the MVPCTPs. Note that typically these heuristic solutions are characterized by high quality objective function values.

5.1. Approximation Model $A1$

The first new Mixed-Integer Linear Programming model is based on the following theoretical results, that produce a direct linear objective function to approximate the original one (2) without the need for additional variables or constraints. The new models could be based on the structure of either model $I2$ or model $I3$. Here we describe them in terms of $I3$, since this was the structure used for the experiments that are outlined in Section 6.

Lemma 1. *Given a binary vector y and a vector γ such that $0 \leq \gamma_i \leq 1 \ \forall i \in V$, the following inequality holds:*

$$\prod_{i \in V}(1 - \gamma_i y_i) \geq 1 - \sum_{i \in V} \gamma_i y_i$$

Proof. If $V = \{1\}$ then the inequality is clearly true. Now let $V = \{1, 2, \ldots, k-1\}$ and assume inductively that

$$\prod_{i \in V}(1 - \gamma_i y_i) \geq 1 - \sum_{i \in V} \gamma_i y_i$$

Then

$$\begin{aligned} \prod_{i \in V \cup \{k\}} (1 - \gamma_i y_i) &\geq (1 - \sum_{i \in V} \gamma_i y_i)(1 - \gamma_k y_k) \\ &= 1 - \sum_{i \in V \cup \{k\}} \gamma_i y_i + \gamma_k y_k \sum_{i \in V} \gamma_i y_i \\ &\geq 1 - \sum_{i \in V \cup \{k\}} \gamma_i y_i \end{aligned}$$

and the result holds by induction. □

The following result, which builds on Lemma 1, allows the definition of a linear objective function that approximates from above, providing therefore a valid upper bound for the cost of the optimal solution of the MVPCTP.

Proposition 1. *Given a solution y, the following inequality holds:*

$$\mathbb{E}(y) = \sum_{j \in W} q_j \left(1 - \prod_{i \in V} (1 - p_{ij})^{y_i}\right) \leq \sum_{i \in V} \left(\sum_{j \in W} q_j p_{ij}\right) y_i$$

Proof.

$$\sum_{j \in W} q_j \left(1 - \prod_{i \in V} (1 - p_{ij})^{y_i}\right) = \sum_{j \in W} q_j \left(1 - \prod_{i \in V} (1 - p_{ij} y_i)\right) =$$

$$= \sum_{j \in W} q_j - \sum_{j \in W} q_j \prod_{i \in V} (1 - p_{ij} y_i) \overset{\text{Lemma 1}}{\leq} \sum_{j \in W} q_j - \sum_{j \in W} q_j \left(1 - \sum_{i \in V} p_{ij} y_i\right) =$$

$$= \sum_{j \in W} q_j - \sum_{j \in W} q_j + \sum_{j \in W} \sum_{i \in V} q_j p_{ij} y_i = \sum_{i \in V} \left(\sum_{j \in W} q_j p_{ij}\right) y_i$$

□

It is now possible to derive the following approximating Mixed Integer Linear Programming model:

$$(A1) \quad \max \sum_{i \in V} \left(\sum_{j \in W} q_j p_{ij}\right) y_i \tag{19}$$

$$s.t. \ (9), (14)\text{–}(18)$$

The new objective function (19) implements the results of Proposition 1. The new model overestimates by construction the expected demand served by an MVPCTP solution, providing in this way a valid upper bound for the cost of the optimal solution of the MVPCTP. This model is simpler to solve than the original non-linear or the tighter linear approximation ones seen in Section 4, since it avoids the extra constraint (12), which is exponential in number. At the same time, each solution of the new model is feasible in terms of the MVPCTP via Equation (1), and its cost can be computed in polynomial time, providing therefore a lower bound for the optimal solution cost.

5.2. Approximation Model A2

A tighter approximation of the objective function (2) than that proposed in Section 5.1 can be achieved at the price of new variables and constraints, as described in the remainder of this section. The following theoretical results are the basis of the new approximation.

Lemma 2. *Given a binary vector y and a vector γ such that $0 \leq \gamma_i \leq 1 \ \forall i \in V$, the following inequality holds:*

$$\prod_{i \in V} (1 - \gamma_i y_i) \geq 1 - \sum_{i \in V} \gamma_i y_i + \sum_{i \in V} \sum_{j \in V, j > i} \gamma_i \gamma_j y_i y_j - \sum_{i \in V} \sum_{j \in V, j > i} \sum_{k \in V, k > j} \gamma_i \gamma_j \gamma_k y_i y_j y_k$$

Proof. If $V = \{1\}$ then the inequality is clearly true. Now let $V = \{1, 2, \ldots, \alpha - 1\}$ and assume inductively that

$$\prod_{i \in V} (1 - \gamma_i y_i) \geq 1 - \sum_{i \in V} \gamma_i y_i + \sum_{i \in V} \sum_{j \in V, j > i} \gamma_i \gamma_j y_i y_j - \sum_{i \in V} \sum_{j \in V, j > i} \sum_{k \in V, k > j} \gamma_i \gamma_j \gamma_k y_i y_j y_k$$

Then

$$\prod_{i \in V \cup \{\alpha\}} (1 - \gamma_i y_i) \geq \left(1 - \sum_{i \in V} \gamma_i y_i + \sum_{i \in V} \sum_{j \in V, j > i} \gamma_i \gamma_j y_i y_j \right.$$
$$\left. - \sum_{i \in V} \sum_{j \in V, j > i} \sum_{k \in V, k > j} \gamma_i \gamma_j \gamma_k y_i y_j y_k \right) (1 - \gamma_\alpha y_\alpha)$$
$$= 1 - \sum_{i \in V \cup \{\alpha\}} \gamma_i y_i + \sum_{i \in V \cup \{\alpha\}} \sum_{j \in V \cup \{\alpha\}, j > i} \gamma_i \gamma_j y_i y_j$$
$$- \sum_{i \in V \cup \{\alpha\}} \sum_{j \in V \cup \{\alpha\}, j > i} \sum_{k \in V \cup \{\alpha\}, k > j} \gamma_i \gamma_j \gamma_k y_i y_j y_k$$
$$+ (\gamma_\alpha y_\alpha) \sum_{i \in V} \sum_{j \in V, j > i} \sum_{k \in V, k > j} \gamma_i \gamma_j \gamma_k y_i y_j y_k$$
$$\geq 1 - \sum_{i \in V \cup \{\alpha\}} \gamma_i y_i + \sum_{i \in V \cup \{\alpha\}} \sum_{j \in V \cup \{\alpha\}, j > i} \gamma_i \gamma_j y_i y_j$$
$$- \sum_{i \in V \cup \{\alpha\}} \sum_{j \in V \cup \{\alpha\}, j > i} \sum_{k \in V \cup \{\alpha\}, k > j} \gamma_i \gamma_j \gamma_k y_i y_j y_k$$

and the result holds by induction. □

The result of Lemma 2 can be exploited by the following proposition to derive a new approximation for the objective function of the MVPCTP.

Proposition 2. *Given a solution y, the following inequality holds:*

$$\mathbb{E}(y) = \sum_{l \in W} q_l \left(1 - \prod_{i \in V} (1 - p_{il})^{y_i}\right) \leq \sum_{i \in V} \left(\sum_{l \in W} q_l p_{il}\right) y_i$$
$$- \sum_{i \in V} \sum_{j \in V, j > i} \left(\sum_{l \in W} q_l p_{il} p_{jl}\right) y_i y_j + \sum_{i \in V} \sum_{j \in V, j > i} \sum_{k \in V, k > j} \left(\sum_{l \in W} q_l p_{il} p_{jl} p_{kl}\right) y_i y_j y_k \quad (20)$$

Proof.

$$\sum_{l \in W} q_l \left(1 - \prod_{i \in V} (1 - p_{ij})^{y_i}\right) = \sum_{l \in W} q_l \left(1 - \prod_{i \in V} (1 - p_{il} y_i)\right)$$
$$= \sum_{l \in W} q_l - \sum_{l \in W} q_l \prod_{i \in V} (1 - p_{il} y_i) \overset{\text{Lemma 2}}{\leq} \sum_{l \in W} q_l - \sum_{l \in W} q_l \left(1 - \sum_{i \in V} p_{il} y_i \right.$$
$$\left. + \sum_{i \in V} \sum_{j \in V, j > i} p_{il} p_{jl} y_i y_j - \sum_{i \in V} \sum_{j \in V, j > i} \sum_{k \in V, k > j} p_{il} p_{jl} p_{kl} y_i y_j y_k \right)$$
$$= \sum_{l \in W} q_l - \sum_{l \in W} q_l + \sum_{l \in W} \sum_{i \in V} q_j p_{ij} y_i - \sum_{l \in W} \sum_{i \in V} \sum_{j \in V, j > i} q_l p_{il} p_{jl} y_i y_j$$
$$+ \sum_{l \in W} \sum_{i \in V} \sum_{j \in V, j > i} \sum_{k \in V, k > j} q_l p_{il} p_{jl} p_{kl} y_i y_j y_k$$
$$= \sum_{i \in V} \left(\sum_{l \in W} q_l p_{il}\right) y_i - \sum_{i \in V} \sum_{j \in V, j > i} \left(\sum_{l \in W} q_l p_{il} p_{jl}\right) y_i y_j$$
$$+ \sum_{i \in V} \sum_{j \in V, j > i} \sum_{k \in V, k > j} \left(\sum_{l \in W} q_l p_{il} p_{jl} p_{kl}\right) y_i y_j y_k$$

□

In order to take advantage of Proposition 2 within a Mixed Integer Programming Model approximating the MVPCTP, new binary variables have to be introduced to capture the quadratic and cubic interactions on y variables appearing in the approximation. The new variable r_{ij} will take value 1 if both facilities i and j are visited, 0 otherwise, and s_{ijk} will take value 1 if facilities i, j and k are all visited, 0 otherwise. The new model is as follows:

$$(A2) \quad \max \sum_{i \in V} \left(\sum_{j \in W} q_j p_{ij} \right) y_i - \sum_{i \in V} \sum_{j \in V, j > i} \left(\sum_{l \in W} q_l p_{il} p_{jl} \right) r_{ij}$$

$$+ \sum_{i \in V} \sum_{j \in V, j > i} \sum_{k \in V, k > j} \left(\sum_{l \in W} q_l p_{il} p_{jl} p_{kl} \right) s_{ijk} \tag{21}$$

$$s.t. \quad (9), (14)\text{–}(18)$$

$$r_{ij} \geq y_i + y_j - 1 \qquad i, j \in V \cup \{0\}, i < j \tag{22}$$

$$3s_{ijk} \leq y_i + y_j + y_k \qquad i, j, k \in V \cup \{0\}, i < j < k \tag{23}$$

$$r_{ij} \in \{0, 1\} \qquad i, j \in V \cup \{0\}, i < j \tag{24}$$

$$s_{ijk} \in \{0, 1\} \qquad i, j, k \in V \cup \{0\}, i < j < k \tag{25}$$

The new objective function (21) translates the results of Proposition 2 in terms of the model. The new constraints (22) and (23) connect variables r and s (capturing quadratic and cubic terms, respectively) to variables y according to the logic and optimization signs of the formulation, while the domain of the new variables is specified by constraints (24) and (25).

Analogously to what has been seen in Section 5.1 for model $A1$, model $A2$ provides valid upper bounds and solutions that, upon evaluation according to (1), provide valid lower bounds.

6. Experimental Results

This section is devoted to the experimental evaluation of the models discussed in Sections 4 and 5. The instances available in the literature are firstly introduced in Section 6.1, while the implementation details for the solvers are provided in Section 6.2. The results of a large campaign of experiments are summarized in Section 6.3 for the exact approaches and in Section 6.4 for the approximation models, respectively. The detailed results obtained by the different models can be found in the Supplementary Material.

6.1. Benchmark Instances

A comprehensive set of instances of the MVPCTP was introduced in [1]. Some well established instances for the Capacitated Vehicle Routing Problem were manipulated and complemented with extra information to obtain a benchmark set for the MVPCTP. Namely a total of 588 instances—ranging from 5 to 161 facilities, and from 10 to 322 customers—were generated, starting from benchmark sets from [27] (set A, 108 instances; set B, 92 instances; set P, 68 instances), [28] (set $E - n$, 44 instances), [29] (set F, 12 instances) and [30] (set E, 264 instances). For each instance, the first one-third of the customers (with rounding to the nearest integer) were converted into facilities, leaving the remaining ones as customers (and the depot), Euclidean distances were rounded to the second decimal position. Probabilities were obtained by first calculating $\hat{p}_{ij} = \min(0.95, \frac{1}{d_{ij}^2})$ $\forall i \in V, j \in W$ (with the assumption that $\hat{p}_{ij} = 0.95$ if $d_{ij} = 0$), then p_{min} and p_{max} were defined as the minimum and the maximum of the calculated values. The probabilities were finally obtained by rounding to the third decimal digit the following quantities: $p_{ij} = p_{min} + (0.95 - p_{min})\frac{\hat{p}_{ij} - p_{min}}{p_{max} - p_{min}}$ $\forall i \in V, j \in W$ (with the assumption that the last fraction takes value 1 when $p_{max} = p_{min}$). The maximum distance was set as two or three times the average distance between facilities and depot, rounded to the closest integer, while the number of vehicles was set to either two or three.

6.2. Implementation Details

As observed in [1], all the models discussed in Sections 4 and 5 can be tightened by changing constraints (6) and (7) to the following ones:

$$t_i \geq d_{0i} y_i \quad i \in V \tag{26}$$

$$t_k \geq \begin{pmatrix} t_i + (L - d_{i0} + d_{ik} - d_{0k}) x_{ik} + \\ +(L - d_{i0} - d_{ki} - d_{0k}) x_{ki} + \\ -(L - d_{i0}) y_i + d_{0k} y_k \end{pmatrix} \quad i, k \in V, k \neq i \tag{27}$$

$$t_i \leq (L - d_{i0}) y_i - (L - d_{i0} - d_{0i}) x_{0i} \leq L \quad i \in V \tag{28}$$

We address the interested reader to [1] for further explanations of the new inequalities. In our experiments, all the models benefited from these enhancements.

We decided to adopt OR-Tools CP-SAT 9.11 [31] as a solver, due to its efficiency in dealing with small/medium TSP-related problems, like the TSP-component of the MVPCTP. Note that similar conclusions have been reached on other problems with similar characteristics (see, for example, ref. [32,33]). To adapt to the characteristics of the solver, the following change was therefore made to model $I3$. Constraints (15) and (16) were excluded and the following one was inserted instead:

$$AddCircuit(z_{ij}^k, (i,j) \in A, i \neq 0 \vee j \neq 0) \quad k \in K \tag{29}$$

Constraint (29) made use of the *AddCircuit* statement provided by the solver adopted, which guaranteed that a feasible circuit was produced by the assignment of the z^k variables. The definition of *AddCircuit* was a perfect match for the constraints it substituted. Notoriously, this statement has the advantage of providing compact formulations and very good performance for small and medium size graphs, at the price of a well known computational burden when the number of vertices of the graph increases too much.

A drawback of the CP-SAT solver currently available is that it is impossible to dynamically add new constraints to the model during the solution process, due to limited callback functionalities. For this reason, it was impossible to efficiently implement into our solver all the valid inequalities mentioned in [1] (and used by the solver discussed there).

While solving models $I2$ and $I3$, inequality (12) is not considered initially, and it is added in an iterative fashion when it is found to be violated by the current solution. In our implementation the model under investigation was iteratively solved to optimality, and in the case that inequality (12) corresponding to the solution found was violated it was added to the model and the process was repeated until there was no violation, which meant that the optimal solution of the MVPCTP instance had been retrieved. Normally, only a few iterations are required, due to the characteristics of the problem and inequality (12). It is however important to observe that the iterative strategy we used is a surrogate of the callback functions of CPLEX, which are much more efficient from a computational viewpoint. Unfortunately the current implementation of CP-SAT does not offer callback functions.

6.3. Linear Models

The models described in Section 4, implemented as described in Section 6.2, were tested on a computer equipped with an Intel Core i7 12700F processor running at 2.1 GHz and 32 GB of RAM, but all the computing times reported, together with the maximum computation time of 3600, were normalized—according to [34] with a factor of 1.27 to express that our processor is faster—to the Intel Xeon Gold 5120 2.20 GHz with 8 GB of RAM adopted for the experiments of [1], in order to have a fair comparison. Note that this normalization did not capture the differences in the performance of the solvers used by the different methods (see below), but since updated (and aligned) versions of the solvers were used for the experiments the results are as fair as possible. With respect to the results originally published in [1], we compared with the results re-computed by the authors of the original paper to overcome some numerical precision issues and kindly communicated to us [35].

The results are summarized in Table 3, where the instances are aggregated by size, by number of vehicles, by deadline and by origin (# being the number of instances in each aggregation). Statistics are reported for the (revised) results of the method described in [1] (adopting CPLEX 22.10 [36] as a solver) and for models $I2$ and $I3$. For each method and aggregation we present the average optimality gap (*Gap* %, calculated as $100 \cdot \frac{UB-LB}{UB}$ where LB and UB are the bounds returned by the approach under investigation), the percentage of instances solved to optimality (*Opt* %), and the average computation time in seconds (*Sec*) for each group of instances. In practical terms, *Gap* % describes the ability of each method to find high-quality solutions, since a small difference between the upper bound (upper estimation of the optimal value) and the lower bound (objective function value of the best solution retrieve) can be read as the inverse of quality confidence. The values in columns *Sec* provide an estimation of the speed and scalability of the different methods, while *Opt* % is another indicator of the effectiveness of the methods, providing the successful rate in proving optimality.

Table 3. Results of the linear models (with a maximum computation time of 3600 s).

Instances		**[1] (rev)**			**I2**			**I3**		
Aggregations	**#**	**Gap %**	**Sec**	**Opt %**	**Gap %**	**Sec**	**Opt %**	**Gap %**	**Sec**	**Opt %**
$\|V\| + \|W\|$										
$[15, 35)$	100	0.000	0.120	100.000	0.000	0.098	100.000	0.000	0.101	100.000
$[35, 40)$	40	0.000	0.447	100.000	0.000	0.144	100.000	0.000	0.136	100.000
$[40, 50)$	76	0.000	1.309	100.000	0.000	0.442	100.000	0.000	0.220	100.000
$[50, 60)$	72	0.000	1.556	100.000	0.000	1.268	100.000	0.000	0.407	100.000
$[60, 70)$	56	0.274	81.215	98.214	0.000	39.219	100.000	0.000	0.895	100.000
$[70, 100)$	92	0.000	52.267	100.000	0.253	156.729	97.826	0.000	2.430	100.000
$[100, 150)$	56	0.071	362.825	94.643	0.537	633.123	89.286	0.002	101.665	98.214
$[150, 200)$	40	0.322	865.908	82.500	5.121	1333.411	65.000	1.026	604.229	87.500
$[200, 300)$	16	7.865	3055.407	18.750	18.177	3062.630	25.000	7.247	2229.237	62.500
$[300, 400)$	28	22.138	3342.235	10.714	28.576	3567.332	3.571	17.304	3147.179	17.857
$[400, 483]$	12	34.284	3600.000	0.000	32.980	3600.000	0.000	19.236	3358.024	16.667
$\|K\|$										
2	294	2.181	388.683	90.816	2.629	461.010	88.435	1.275	275.729	93.878
3	294	1.865	462.422	88.435	3.306	551.537	86.735	1.693	385.096	90.816
L										
2	294	1.535	321.659	92.517	1.720	318.156	92.177	0.608	223.822	95.578
3	294	2.510	529.446	86.441	4.215	694.391	82.993	2.360	437.003	89.116
Set										
A	108	0.000	5.392	100.000	0.216	68.941	98.148	0.000	0.900	100.000
B	92	0.167	44.729	98.913	0.000	21.679	100.000	0.000	0.511	100.000
E	264	4.444	894.342	77.652	6.429	1044.140	73.485	3.305	730.109	82.955
E-n	44	0.000	17.104	100.000	0.000	118.075	100.000	0.000	1.708	100.000
F	12	0.067	681.571	91.667	0.432	386.748	91.667	0.000	47.886	100.000
P	68	0.000	7.200	100.000	0.000	38.816	100.000	0.000	0.638	100.000
All	588	2.023	425.552	89.626	2.967	506.273	87.925	1.484	330.413	92.347

The results of Table 3 lead to a few considerations. First, since the solver discussed in [1] and $I2$ are based on the same ideas, the results suggest that the use of CPLEX callbacks and of the extra-valid inequalities we did not implement are instrumental in obtaining better results. On the other hand, model $I3$ obtained better results than the other solvers, showing that the new representation of the vehicle routing component we propose has a role. In particular, all the indicators suggested the superiority of $I3$ both in terms of computation speed and the quality of the results obtained. The aggregation on the size of the instances ($|V| + |W|$) also indicates a substantially better scalability. Independently of the model, the table also indicates that instances with more vehicles or larger deadlines are more difficult to solve. It can also be observed that all the instances not solved to optimality belonged to the set E, which contained the larger instances. Finally, although this does not emerge directly from the statistics presented here, some statistics can be done when the results achieved are compared with the previous state-of-the-art solutions available in the literature. Solving model $I2$ led to 32 improved lower bounds and 8 improved upper bounds (see Appendix A), suggesting that the CP-SAT solver performs very well on this particular model; solving model $I3$ led to 41 improved lower bounds and 38 improved upper bounds (see Appendix B); 20 instances were closed for the first time (see Appendix D).

6.4. Approximation Models

The approximation models discussed in Section 5 were tested on the benchmark instances available from the literature, with a maximum computation time normalized to 360 s on the Intel Xeon Gold 5120 2.20 GHz computer with 8 GB of RAM adopted for the experiments in [1], therefore with one tenth of the time allotted to the linear models (see Section 6.3). The shorter time was justified by the fact that models $A1$ and $A2$ had to be solved only once to output the approximation, with no need for iteratively added inequalities (unlike the linear models).

The results of the experiments are summarized in Table 4, where the instances are aggregated according to the same criteria used for Table 3. For the two approximation methods $A1$ and $A2$, the following statistics are reported. For both the lower and the upper bounds provided, the percentage deviation from the best known bound was calculated as $100 \cdot \frac{|Best - New|}{Best}$, where *Best* was the respective best-known bound from Table 3 and *New* was the bound provided by the approximation model. Moreover, the number of improved bounds (leading to new state-of-the-art results) is reported. In the Table, the results of model $I3$ with a maximum computation time of 360 s are also reported as a reference.

Table 4 clearly indicates that the exact model $I3$ had problems converging on the large instances with only 360 s available. In these cases both the lower and the upper bounds presented substantial gaps. The following considerations are in order for the approximation models. First, model $A1$ was able to capture well the essence of the problem, being able to provide very high-quality heuristic solutions (lower bounds) in short time, with an average deviation below 0.2 % from the best-known results. There was a general degradation of performance in many of the large instances, indicating that the computation time allotted was probably not enough in those instances for the given approach. There were, however, exceptions in the largest instances, for which some of the bounds were actually improved in seven instances (see Table A5), indicating that model $I3$ is likely to suffer from scalability issues. There was even a negative average deviation for the largest instances, indicating that the approach is consistently better than the methods previously available. Concerning the upper bounds provided by method $A1$, they were looser than the lower bounds, indicating that the approximation of the objective function was not precise enough. Secondly, model $A2$ was able to provide optimal solutions and small deviations from the upper bounds for small instances; then the extra-complexity added (in terms of variables and constraints) to improve the approximation made the model too heavy in the larger instances, leading to poor results. Thirdly, all the improved solutions found were for instances with larger deadlines (L), indicating that in some of these instances there was probably a greater margin for improvements. Finally, it is worth noting that, thanks to an improved lower bound, one more instance was closed to optimality for the first time in this study (see Table A6).

Table 4. Results of the approximation models (with a maximum computation time of 360 s).

Instances		I3		A1				A2			
Aggregations	#	LB	UB	LB		UB		LB		UB	
		Dev %	Dev %	Dev %	# Impr	Dev %	# Impr	Dev %	# Impr	Dev %	# Impr
$\|V\| + \|W\|$											
$[15, 35)$	100	0.000	0.000	0.076	0	3.285	0	0.000	0	0.029	0
$[35, 40)$	40	0.000	0.000	0.187	0	7.138	0	0.000	0	0.243	0
$[40, 50)$	76	0.000	0.000	0.107	0	8.110	0	0.000	0	0.295	0
$[50, 60)$	72	0.000	0.000	0.066	0	10.240	0	0.000	0	0.118	0
$[60, 70)$	56	0.000	0.000	0.017	0	5.672	0	0.000	0	0.039	0
$[70, 100)$	92	0.000	0.000	0.030	0	13.289	0	0.000	0	0.545	0
$[100, 150)$	56	0.011	0.267	0.366	0	11.974	0	0.172	0	3.343	0
$[150, 200)$	40	0.213	3.319	0.170	0	7.133	0	1.919	0	10.125	0
$[200, 300)$	16	6.718	37.681	2.227	0	32.080	0	6.921	0	48.400	0
$[300, 400)$	28	32.345	59.955	1.068	3	24.818	0	11.940	0	51.543	0
$[400, 483]$	12	29.625	82.451	−1.150	4	30.850	0	28.750	0	96.255	0
$\|K\|$											
2	294	2.480	7.414	0.162	4	11.227	0	1.327	0	7.990	0
3	294	2.207	4.215	0.215	3	9.324	0	1.654	0	5.823	0
L											
2	294	2.080	5.956	0.159	0	8.166	0	0.652	0	5.689	0
3	294	2.607	5.672	0.217	7	12.385	0	2.329	0	8.124	0
Set											
A	108	0.000	0.000	0.019	0	3.826	0	0.000	0	0.308	0
B	92	0.000	0.000	0.025	0	3.512	0	0.000	0	0.023	0
E	264	5.219	12.929	0.338	7	12.765	0	3.312	0	14.887	0
E-n	44	0.000	0.000	0.114	0	14.694	0	0.000	0	0.783	0
F	12	0.000	0.456	0.670	0	6.980	0	0.174	0	1.895	0
P	68	0.000	0.000	0.061	0	17.730	0	0.000	0	0.561	0
All	588	2.343	5.814	0.188	7	10.276	0	1.491	0	6.906	0

In conclusion, model $A2$ appeared to be dominated by $I3$ on runs of 360 s, since both these models seemed to suffer from scalability issues. When high-quality solutions had to be retrieved in a short time for the larger instances, model $A1$ was clearly the best option.

7. Conclusions

New models to address the Multi-Vehicle Probabilistic Covering Tour Problem have been presented. In this problem, a fleet of vehicles characterized by a limited range has to distribute goods to some given facilities. Once the goods are at the facilities they are successfully dispatched to the final customers according to a given probability. The target is to plan a set of routes for the trucks that maximizes the expected quantity of goods delivered to the final customers.

A new linear model was introduced and shown to lead to several new state-of-the-art results. Finally, new approximation models, able to provide high-quality solutions and bound in short time, and shown to be very effective in larger instances, were introduced and experimentally validated.

The new methods we propose are intrinsically easy to implement, and they are without parameter tuning, making them ideal for companies looking for effective methods easy to control. We propose models that are able to match and improve state-of-the-art results, but which require more computation time, and approximation ones that run substantially quicker but are still able to provide high-quality solutions. These tools can be easily implemented by companies facing the Multi-Vehicle Probabilistic Covering Tour Problem (or its variations, with some adjustments), improving their operative strategies. The tools are also of interest for the disaster management sector, where conceptually simple methods are important for adaptability to particular situations.

Supplementary Materials: The extended results of the algorithms described in the paper can be downloaded at: https://www.mdpi.com/article/10.3390/a19080643/s1.

Author Contributions: Conceptualization, R.M. and D.H.S.; methodology, R.M.; software, R.M.; validation, R.M. and D.H.S.; formal analysis, R.M. and D.H.S.; investigation, R.M. and D.H.S.; resources, R.M.; data curation, R.M.; writing—original draft preparation, R.M. and D.H.S.; writing—review and editing, R.M. and D.H.S.; visualization, R.M. and D.H.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The instances used for the experiments are available upon request to the authors.

Acknowledgments: The authors are grateful to Günes Erdoğan and İsmail Karaoğlan for useful discussions and suggestions.

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

Appendix A. Bounds Improved by Model *I*2 over the Previous Best-Known Results

In Tables A1 and A2 the solutions improved by solving model *I*2 are summarized. The values of the bounds previously available in the literature (baseline) and found by solving model *I*2 are provided.

Table A1. The 32 lower bounds improved by model *I*2 over the previous best-known results.

Instance Name	LB from [1] (rev)	LB Found by Solving *I*2
E200-16b(3_79).mexctp	418.063	421.267
E200-17b(3_79).mexctp	418.063	421.267
E241-22k(2_23).mexctp	693.533	694.126
E241-22k(3_23).mexctp	952.931	961.720
E253-27k(2_15).mexctp	2177.920	2365.899
E253-27k(2_23).mexctp	4146.740	4245.529
E253-27k(3_15).mexctp	2768.470	3189.182
E256-14k(2_18).mexctp	996.697	1003.442
E301-28k(2_24).mexctp	735.389	757.098
E301-28k(3_24).mexctp	1004.570	1049.451

Table A1. *Cont.*

Instance Name	LB from [1] (rev)	LB Found by Solving $I2$
E321-30k(2_17).mexctp	2085.590	2428.988
E321-30k(2_25).mexctp	3682.040	3881.432
E321-30k(3_17).mexctp	3073.380	3351.320
E321-30k(3_25).mexctp	4980.450	5242.036
E324-16k(2_21).mexctp	1085.870	1091.349
E324-16k(2_32).mexctp	2455.060	2467.983
E361-33k(2_25).mexctp	639.409	716.066
E361-33k(3_25).mexctp	922.290	1033.636
E386-47t(2_538).mexctp	117.068	118.577
E386-47t(2_808).mexctp	138.424	140.492
E397-34k(2_18).mexctp	1980.580	2095.273
E397-34k(3_18).mexctp	2687.680	2930.816
E400-18k(2_35).mexctp	2353.790	2358.186
E400-18k(3_23).mexctp	1253.900	1357.975
E400-18k(3_35).mexctp	2941.610	3079.991
E421-41k(2_26).mexctp	544.330	637.795
E481-38k(2_20).mexctp	2465.700	2730.506
E481-38k(3_20).mexctp	2940.940	3828.439
E481-38k(3_30).mexctp	5406.910	6637.113
E484-19k(2_26).mexctp	989.263	1052.885
E484-19k(3_26).mexctp	1384.760	1444.954
E484-19k(3_39).mexctp	2979.550	2991.120

Table A2. The 8 upper bounds improved by model $I2$ over the previous best-known results.

Instance Name	UB from [1] (rev)	UB Found by Solving $I2$
E253-27k(2_15).mexctp	2970.020	2365.899
E253-27k(3_15).mexctp	4154.260	3839.901
E321-30k(2_17).mexctp	2960.360	2882.145
E400-18k(2_23).mexctp	1631.020	1477.119
E421-41k(2_17).mexctp	347.385	344.254
E421-41k(2_26).mexctp	1236.660	1234.612
E481-38k(2_30).mexctp	7064.860	4255.634
E484-19k(2_26).mexctp	1681.940	1569.567

Appendix B. Bounds Improved by Model $I3$ over the Previous Best-Known Results

In Tables A3 and A4 the solutions improved by solving model $I3$ are summarized. The values of the bounds previously available in the literature (baseline) and found by solving model $I3$ are provided.

Table A3. The 41 lower bounds improved by model $I3$ over the previous best-known results.

Instance Name	LB from [1] (rev)	LB Found by Solving $I3$
E200-17b(3_79).mexctp	418.063	421.267
E241-22k(2_23).mexctp	693.533	694.611
E241-22k(3_23).mexctp	952.931	959.412
E253-27k(2_15).mexctp	2177.920	2365.899
E253-27k(2_23).mexctp	4146.740	4253.261
E253-27k(3_15).mexctp	2768.470	3229.921
E253-27k(3_23).mexctp	5573.140	5643.656
E256-14k(2_18).mexctp	996.697	1003.442
E262-25j(3_239).mexctp	393.255	401.413
E301-28k(2_24).mexctp	735.389	765.302
E301-28k(3_24).mexctp	1004.570	1056.084

Table A3. *Cont.*

Instance Name	LB from [1] (rev)	LB Found by Solving $I3$
E321-30k(2_17).mexctp	2085.590	2428.988
E321-30k(2_25).mexctp	3682.040	4161.569
E321-30k(3_17).mexctp	3073.380	3351.320
E321-30k(3_25).mexctp	4980.450	5246.425
E324-16k(2_21).mexctp	1085.870	1091.349
E324-16k(2_32).mexctp	2455.060	2531.097
E324-16k(3_21).mexctp	1499.670	1501.195
E324-16k(3_32).mexctp	3097.470	3188.641
E361-33k(2_25).mexctp	639.409	779.251
E361-33k(3_25).mexctp	922.290	1049.203
E386-47t(2_538).mexctp	117.068	118.718
E386-47t(2_808).mexctp	138.424	141.919
E386-47t(3_538).mexctp	121.479	121.867
E397-34k(2_18).mexctp	1980.580	2095.273
E397-34k(2_27).mexctp	3677.060	3688.606
E397-34k(3_18).mexctp	2687.680	2940.207
E397-34k(3_27).mexctp	4549.380	4970.547
E400-18k(2_23).mexctp	1024.230	1025.608
E400-18k(2_35).mexctp	2353.790	2477.875
E400-18k(3_23).mexctp	1253.900	1421.717
E400-18k(3_35).mexctp	2941.610	2997.808
E421-41k(2_26).mexctp	544.330	631.534
E421-41k(3_26).mexctp	769.464	902.101
E481-38k(2_20).mexctp	2465.700	2730.506
E481-38k(3_20).mexctp	2940.940	3828.561
E481-38k(3_30).mexctp	5406.910	6637.113
E484-19k(2_26).mexctp	989.263	1072.053
E484-19k(2_39).mexctp	2389.150	2486.199
E484-19k(3_26).mexctp	1384.760	1459.812
E484-19k(3_39).mexctp	2979.550	3296.590

Table A4. The 38 upper bounds improved by model $I3$ over the previous best-known results.

Instance Name	UB from [1] (rev)	UB Found by Solving $I3$
B-n68-k9(2_189).mexctp	123.655	104.667
E121-07c(2_198).mexctp	57.701	56.356
E135-07f(2_49).mexctp	514.471	514.470
E135-07f(3_73).mexctp	2200.370	2182.152
E151A15r(3_271).mexctp	771.946	767.075
E151B14r(2_218).mexctp	328.394	327.227
E151D14r(3_196).mexctp	593.563	589.156
E241-22k(2_23).mexctp	768.425	761.126
E253-27k(2_15).mexctp	2970.020	2365.899
E253-27k(2_23).mexctp	4965.120	4840.706
E253-27k(3_15).mexctp	4154.260	3229.921
E256-14k(2_18).mexctp	1085.230	1003.442
E256-14k(3_18).mexctp	1386.460	1345.691
E262-25j(2_239).mexctp	332.057	329.540
E262-25j(3_159).mexctp	299.829	295.501
E262-25j(3_239).mexctp	414.441	401.413
E301-28k(2_24).mexctp	1168.480	1072.589
E321-30k(2_17).mexctp	2960.360	2428.988
E321-30k(3_17).mexctp	4045.030	3351.320
E324-16k(2_21).mexctp	1451.960	1239.335
E361-33k(2_25).mexctp	1342.110	1005.921
E361-33k(3_25).mexctp	1586.450	1404.373

https://doi.org/10.3390/a19080643

Table A4. *Cont.*

Instance Name	UB from [1] (rev)	UB Found by Solving $I3$
E386-47t(2_538).mexctp	122.476	118.718
E397-34k(2_18).mexctp	2641.550	2095.273
E397-34k(2_27).mexctp	5881.970	5656.976
E397-34k(3_18).mexctp	3793.580	2940.207
E400-18k(2_23).mexctp	1631.020	1150.409
E400-18k(3_23).mexctp	2045.390	1842.815
E421-41k(2_17).mexctp	347.385	334.027
E421-41k(2_26).mexctp	1236.660	844.117
E421-41k(3_26).mexctp	1467.680	1232.353
E481-38k(2_20).mexctp	3510.780	2730.506
E481-38k(2_30).mexctp	7064.860	4255.634
E481-38k(3_20).mexctp	4981.600	3957.877
E484-19k(2_26).mexctp	1681.940	1402.352
E484-19k(3_26).mexctp	2124.180	2027.122
F-n135-k7(2_49).mexctp	514.471	514.470
F-n135-k7(3_73).mexctp	2199.790	2182.152

Appendix C. Bounds Improved by the Approximation Model $A1$ over the Best-Known Results from the Previous Literature and Models $I2$ and $I3$

In Table A5 the lower bounds improved by solving the approximation model $A1$ are summarized. The values of the best lower bounds available from the previous literature and those found by solving models $I2$ and $I3$ are used as a baseline.

Table A5. The 7 lower bounds improved by model $A1$ over the previous literature and models $I2$ and $I3$.

Instance Name	Best LB from [1] (rev), $I2$ and $I3$	LB Found by Solving $A1$
E361-33k(3_25).mexctp	1049.203	1092.903
E386-47t(3_808).mexctp	145.293	145.644
E397-34k(2_27).mexctp	3688.606	3949.449
E421-41k(2_26).mexctp	637.795	680.161
E421-41k(3_26).mexctp	902.101	902.530
E481-38k(2_30).mexctp	4255.630	4255.634
E484-19k(2_39).mexctp	2486.199	2513.556

Appendix D. Instances Closed for the First Time in This Study

The instanced for which an optimal solution has been proven for the first time by the present study are summarized in Table A6. The best lower and upper bounds available from the previous literature are reported together with the proven optimal value. Note that all the instances apart from *E481-38k(2_30).mexctp* were closed by the improved bounds provided by solving models $I2$ and $I3$ (see Section 4), while *E481-38k(2_30).mexctp* was closed by the improved lower bound provided by solving the approximation model $A1$ (see Section 5.1).

Table A6. The 21 instances closed for the first time in the present study.

Instance Name	Best Bounds from [1] (rev)		Optimal Value
B-n68-k9(2_189).mexctp	104.667	123.655	104.667
E121-07c(2_198).mexctp	56.356	57.701	56.356
E135-07f(3_73).mexctp	2182.150	2200.370	2182.152
E151A15r(3_271).mexctp	767.077	771.946	767.075
E151B14r(2_218).mexctp	327.227	328.394	327.227
E253-27k(2_15).mexctp	2177.920	2970.020	2365.899

Table A6. *Cont.*

Instance Name	Best Bounds from [1] (rev)		Optimal Value
E253-27k(3_15).mexctp	2768.470	4154.260	3229.921
E256-14k(2_18).mexctp	996.697	1085.230	1003.442
E256-14k(3_18).mexctp	1345.690	1386.460	1345.691
E262-25j(2_239).mexctp	329.540	332.057	329.540
E262-25j(3_159).mexctp	295.502	299.829	295.501
E262-25j(3_239).mexctp	393.255	414.441	401.413
E321-30k(2_17).mexctp	2085.590	2960.360	2428.988
E321-30k(3_17).mexctp	3073.380	4045.030	3351.320
E386-47t(2_538).mexctp	117.068	122.476	118.718
E397-34k(2_18).mexctp	1980.580	2641.550	2095.273
E397-34k(3_18).mexctp	2687.680	3793.580	2940.207
E421-41k(2_17).mexctp	334.028	347.385	334.027
E481-38k(2_20).mexctp	2465.700	3510.780	2730.506
E481-38k(2_30).mexctp	4255.630	7064.860	4255.634
F-n135-k7(3_73).mexctp	2182.150	2199.790	2182.152

References

1. Karaoğlan, İ.; Erdoğan, G.; Koç, Ç. The Multi-Vehicle Probabilistic Covering Tour Problem. *Eur. J. Oper. Res.* **2018**, *271*, 278–287. [CrossRef]
2. Gendreau, M.; Laporte, G.; Semet, F. The covering tour problem. *Oper. Res.* **1997**, *45*, 568–576. [CrossRef]
3. Altay, N.; Green, W.G. OR/MS research in disaster operations management. *Eur. J. Oper. Res.* **2006**, *175*, 475–493. [CrossRef]
4. De La Torre, L.E.; Dolinskaya, I.S.; Smilowitz, K.R. Disaster relief routing: Integrating research and practice. *Socio-Econ. Plan. Sci.* **2012**, *46*, 88–97. [CrossRef]
5. Pajares, G. Overview and current status of remote sensing applications based on unmanned aerial vehicles (UAVs). *Photogramm. Eng. Remote Sens.* **2015**, *81*, 281–329. [CrossRef]
6. Toth, P.; Vigo, D. *Vehicle Routing: Problems, Methods, and Applications*; MOS-SIAM Series on Optimization: Philadelphia, PA, USA, 2014.
7. Gendreau, M.; Jabali, O.; Rei, W. Future research directions in stochastic vehicle routing. *Transp. Sci.* **2016**, *50*, 1163–1173. [CrossRef]
8. Vargas, L.; Jozefowiez, N.; Ngueveu, S.U. A selector operator-based adaptive large neighborhood search for the covering tour problem. In *Learning and Intelligent Optimization*; Springer: Cham, Switzerland, 2015; pp. 170–185.
9. Jozefowiez, N.; Semet, F.; Talbi, E. The bi-objective covering tour problem. *Comput. Oper. Res.* **2007**, *34*, 1929–1942. [CrossRef]
10. Hachicha, M.; Hodgson, M.J.; Laporte, G.; Semet, F. Heuristics for the multi-vehicle covering tour problem. *Comput. Oper. Res.* **2000**, *27*, 29–42. [CrossRef]
11. Lopes, R.; Souza, V.A.A.; da Cunha, A.S. A branch-and-price algorithm for the multi-vehicle covering tour problem. *Electron. Notes Discret. Math.* **2013**, *44*, 61–66. [CrossRef]
12. Jozefowiez, N. A branch-and-price algorithm for the multi-vehicle covering tour problem. *Networks* **2014**, *64*, 160–168. [CrossRef]
13. Ota, C.T.; Fiorotto, D.J.; Ghidini, C.T.L.d.S.; Oliveira, W.A.d. A flow-based model for the multivehicle covering tour problem with route balancing. *Int. Trans. Oper. Res.* **2024**, *31*, 2930–2955.
14. Margolis, J.T.; Song, Y.; Mason, S.J. A multi-vehicle covering tour problem with speed optimization. *Networks* **2022**, *79*, 119–142.
15. Hà, M.H.; Bostel, N.; Langevin, A.; Rousseau, L.M. An exact algorithm and a metaheuristic for the multi-vehicle covering tour problem with a constraint on the number of vertices. *Eur. J. Oper. Res.* **2013**, *226*, 211–220. [CrossRef]
16. Glize, E.; Roberti, R.; Jozefowiez, N.; Ulrich Ngueveu, S. Exact Methods for MonoObjective and Bi-Objective Multi-Vehicle Covering Tour Problems. *Eur. J. Oper. Res.* **2020**, *283*, 812–824. [CrossRef]
17. Allahyari, S.; Salari, M.; Vigo, D. A hybrid metaheuristic algorithm for the multi-depot covering tour vehicle routing problem. *Eur. J. Oper. Res.* **2015**, *242*, 756–768. [CrossRef]
18. Flores-Garza, D.A.; Salazar-Aguilar, M.A.; Ngueveu, S.U.; Laporte, G. The multi-vehicle cumulative covering tour problem. *Ann. Oper. Res.* **2017**, *258*, 761–780. [CrossRef]
19. Pham, T.A.; Há, M.H.; Nguyen, X.H. Solving the multi-vehicle multi-covering tour problem. *Comput. Oper. Res.* **2017**, *88*, 258–278. [CrossRef]
20. Kammoun, M.; Derbel, H.; Jarboui, B. Two meta-heuristics for solving the multi-vehicle multi-covering tour problem with a constraint on the number of vehicles. *Yugosl. J. Oper. Res.* **2021**, *31*, 299–318. [CrossRef]

21. Oliveira, B.; Pessoa, A.; Roboredo, M. New cuts and a branch-cut-and-price model for the multi-vehicle covering tour problem. *4OR* **2025**, *23*, 225–245. [CrossRef]
22. Fischer, V.; Pacheco Paneque, M.; Legrain, A.; Bürgy, R. A capacitated multi-vehicle covering tour problem on a road network and its application to waste collection. *Eur. J. Oper. Res.* **2024**, *315*, 338–353. [CrossRef]
23. Hagn, A.; Krause, J.; Stargalla, M.; Moreno, L. An Adaptive Variable Neighborhood Search for a Family of Set Covering Routing Problems with an Application in Disaster Relief Operations. *arXiv* **2026**. [CrossRef]
24. Montemanni, R.; Smith, D.H. A model-based algorithm for the Probabilistic Orienteering Problem. *Comput. Oper. Res.* **2025**, *176*, 106947. [CrossRef]
25. Montemanni, R.; Smith, D.H. Model-based algorithms for the 0-1 Time-Bomb Knapsack Problem. *Comput. Oper. Res.* **2025**, *178*, 107010. [CrossRef]
26. Dantzig, G.; Fulkerson, R.; Johnson, S. Solution of a Large Scale Traveling Salesman Problem. *Oper. Res.* **1954**, *2*, 393–410. [CrossRef]
27. Augerat, P.; Naddef, D.; Belenguer, J.M.; Benavent, E.; Corberan, A.; Rinaldi, G. *Computational Results with a Branch and Cut Code for the Capacitated Vehicle Routing Problem*; Technical Report Research Report 949-M; Université Joseph Fourier: Grenoble, France, 1995.
28. Christofides, N.; Eilon, S. An Algorithm for the Vehicle-dispatching Problem. *J. Oper. Res. Soc.* **1969**, *20*, 309–318. [CrossRef]
29. Fisher, M.L. Optimal Solution of Vehicle Routing Problems Using Minimum K-Trees. *Oper. Res.* **1994**, *42*, 626–642. [CrossRef]
30. VRPLIB. A Vehicle Routing Problem Library-Symmetric CVRP Instances. 1998. Available online: http://or.dei.unibo.it/library/vrplib-vehicle-routing-problem-library (accessed on 30 July 2026).
31. Perron, L.; Didier, F. Google OR-Tools— CP-SAT. 2026. Available online: https://developers.google.com/optimization/cp/cp_solver/ (accessed on 30 July 2026).
32. Montemanni, R.; Dell'Amico, M.; Corsini, A. Parallel drone scheduling vehicle routing problems with collective drones. *Comput. Oper. Res.* **2024**, *163*, 106514. [CrossRef]
33. Montemanni, R.; Ceschia, S.; Schaerf, A. A compact model for the home healthcare routing and scheduling problem. *EURO J. Comput. Optim.* **2025**, *13*, 100101. [CrossRef]
34. PassMark Software Pty Ltd. CPU Benchmarks. 2026. Available online: https://www.cpubenchmark.net/ (accessed on 30 July 2026).
35. Karaoğlan, İ. (Konya Technical University, Konya, Turkey); Erdoğan, G. (University of Bath, Bath, UK). Personal communication, 2024.
36. IBM. IBM CPLEX Optimizer. 2026. Available online: https://www.ibm.com/products/ilog-cplex-optimization-studio/cplex-optimizer (accessed on 30 July 2026).