Journal of Mathematical Modeling
Vol. 13, No. 4, 2025, pp. 747-761. Research Article

JMM

A mixed integer linear programming model for vehicle routing problem for non-complete graphs: Behshahr (Iran) case study

Maria Afsharirad*

Department of Mathematics, University of Science and Technology of Mazandaran, P.O. Box: 48518-78195, Behshahr, Iran
Email(s): m.afsharirad@mazust.ac.ir

Abstract. We consider Vehicle Routing Problem (VRP) for non-complete graphs. In order to avoid converting all networks to complete graphs, as in travelling salesman problem, we model VRP for non-complete graphs. Subtours are allowed in this model, since they are unavoidable in non-complete structure, while disconnected subtours are not allowed. Since disconnected subtour elimination constraints are time-consuming, we provide a separation problem for these constraints and provide an extended formulation based on this separation problem. This extended formulation turns to be equivalent to the original model. In order to reduce the size of the graph, a blocking procedure is proposed in this paper. In addition, we provide two types of valid inequalities to strengthen the formulation. Finally we test our model on a real case study and compare it to the classical model for complete graphs.

Keywords: Vehicle routing problem, extended formulation, disconnected subtour elimination constraints, separation problem, blocking procedure.
AMS Subject Classification 2010: 90C10, 90C11.

1 Introduction

The Vehicle Routing Problem (VRP) is a cornerstone of combinatorial optimization, focusing on the optimal routing of a fleet of vehicles to service a set of customers. Traditionally, VRP is modeled on complete graphs, where each pair of nodes is connected by an edge. However, many real-world applications involve non-complete graphs, necessitating specialized formulations and solution approaches.
Mixed integer formulations have been extensively employed to model VRPs. These formulations typically involve binary variables representing the presence of arcs in the solution and continuous variables

*Corresponding author
Received: 12 January 2025 / Revised: 21 April 2025 / Accepted: 26 April 2025
DOI: 10.22124/jmm.2025.29552.2627

for flow constraints. In the context of non-complete graphs, Mixed Integer Problem (MIP) must account for the absence of certain arcs, which can complicate the model. The classical two-index vehicle flow formulation, an extension of the Traveling Salesman Problem (TSP) formulation by Dantzig et al., [6], has been adapted for VRPs. This formulation uses binary variables associated with each arc to indicate whether it is traversed by a vehicle. However, in non-complete graphs, the absence of arcs requires modifications to ensure feasibility and optimality. The main reason of assuming a complete graph as an input is that, such a structure lets us to force the routes to visit every node exactly once, without being concerned about infeasibilities. No subtours should be generated in this structures. However, subtour elimination constraints are time consuming.

To the best of our knowledge, there is no mathematical model for VRP or even for TSP on non-complete graphs. In this paper we provide an MIP for a directed non-complete graph. This structure avoids solving several shortest path problems and reduces the graph size we are working with. Subtour elimination constraints make our model infeasible. Therefore, we provide a separation problem in order to find feasible routes in a non-complete directed graph and extend the model.

Extended formulations aim to represent combinatorial optimization problems more compactly by introducing additional variables and constraints, potentially reducing the problem's complexity. For VRPs on non-complete graphs, extended formulations can capture the problem's structure more effectively, leading to improved solution methods. Research in [3] discusses the size of perfect formulations for combinatorial optimization problems, providing insights into constructing efficient extended formulations.

VRP was first stated in [6] to deliver gasoline to service stations. Afterwards, various versions of VRP were extended, Capacitated VRP and VRP with Time Windows (VRPTW) are most applicable versions of the problem. Authors in [9] show that all versions of VRP are *NP*-hard. Various MIPs have been proposed for different versions of VRP. For the most recent surveys on exact methods on VRP see [8] and also [11]. Generally, there are two types of formulations in literature. In both types a complete graph is given as an input. The first type is vehicle flow formulation, in which binary variables are associated to arcs of a network representation of the problem. This type of formulation has a weak linear relaxation, even if strengthening with valid inequalities and applying branch-and-cut method. Authors in [16] provide an integer linear programming for CVRP and eliminate subtours applying branch and cut algorithm. The second type which provides a stronger linear relaxation is set partitioning formulation, with smaller number of constraints, but huge number of variables with respect to vehicle flow formulation. Authors in [12] show that both formulations are p-step formulations with particular choices of p. They also apply column generation to solve a p-step formulation. See [5] for an elaborated review on branch-price and cut algorithms on VRP.

According to *NP*-hardness of the problem, several heuristics and meta-heuristics are also applied in literature to approximately solve VRP. Authors in [15] provide a heuristic-based parthenogenetic algorithm (HPGA) to solve VRPTW. A heuristic method based on a variant of *K*-median problem is proposed in [2]. A clustering-based heuristic is proposed in [1]. See [10] for the most recent review on heuristic methods for VRP.

Matheuristics, which combine mathematical programming and metaheuristic approaches, have emerged as a powerful framework for solving VRP. These hybrid techniques leverage the exactness of mathematical models and the flexibility of heuristics to tackle the complexity of VRP variants effectively. Recent studies highlight the use of matheuristics in addressing large-scale and real-world VRPs, including dynamic and stochastic variations, through strategies such as decomposition, column generation, and local search, [7]. Prominent applications include the integration of branch-and-bound with heuristic methods and the use of Lagrangian relaxation to guide heuristic search. This synergy has proven particularly effective in scenarios involving tight constraints or high-dimensional decision spaces, enabling robust solutions with improved computational efficiency, [13].

We model VRP for Non-complete Graphs (VRPNG). It is clear that VRP with a single vehicle without capacity limitation would turn to the TSP. TSP is also defined on a complete graph. As far as we know, the only research on TSP on non-complete graphs has been carried out by [14]. She studied six algorithms in which a TSP tour and a subset of edges in the complete graph are given and the output is a TSP tour that connects the same cities with a shorter length. All these algorithms are based on approximate methods to generate near-optimal solutions.

This paper is organized as follows. Section 2 describes VRPNG and provides a MIP for that. Section 3 discusses an extended formulation for VRPNG. Section 4 provides a heuristic method to reduce the network size and adjust the model for the reduced graph. Section 5 presents two valid inequalities. Finally, section 6 reports the results of computational experiments on a real case study.

2 Problem statement

The problem addressed in this paper consists of designing efficient routes for K identical vehicles to service a set of customers in a non-complete graph. This paper focuses on VRPNG, in which any vehicle is allowed to visit a customer more than once but serving any customer is exactly done in one visit by one vehicle.

2.1 MIP model

Now we provide an MIP for VRPNG. Consider a set of K identical vehicles $U = \{u_1, \dots, u_K\}$, where u_k for $k = 1, \dots, K$ represents a vehicle of capacity C. Every vehicle starts its route from parking and services a set of customers according to their demands and finally ends its route to terminal. Customers are assumed in a directed non-complete graph $G = (V, A)$, where $V = \{S, 1, \dots, n, T\}$ is the set of nodes and $\{1, \dots, n\}$ is the set of all customers with demand q_i for customer $i = 1, \dots, n$. $S \in V$ and $T \in V$ are source and terminal nodes, respectively. Also, $A \subset V \times V$ is the set of edges and c_{ij} is the length of arc $(i, j) \in A$. Any customer should be serviced exactly once by one vehicle, while it might be visited more than once by several vehicles.

Despite existing VRP models for complete graphs, here we cannot eliminate subtours any more, since it may lead to infeasiblity. Figure 1 shows a non-complete graph for 2 vehicles with 3 units of capacity, in which subtour is unavoidable. Note that all arcs in Figure 1 are bi-directed. The first vehicle's path is $(S, 1, 2, T)$, and the second one's is $(S, 1, 3, 5, 4, 3, 1, 2, T)$, where $(3, 5, 4, 3)$ is a subtour in the second route.

As a result, we should allow any node to be visited more than once but serving the node should be done exactly once, also there should be a path from source node S to all other nodes, in any route. More exactly subtours are allowed but disconnected ones are not. Figure 2 shows a disconnected subtour in the graph which is not allowed in our model and should be eliminated. Solid and dotted directed arcs show the first and the second route respectively. The second route is not allowed since it contains a disconnected subtour. Moreover, it should be noted that in a non-complete directed graph, any arc (i, j) may be passed more than once. Figure 3 is an example of this case.

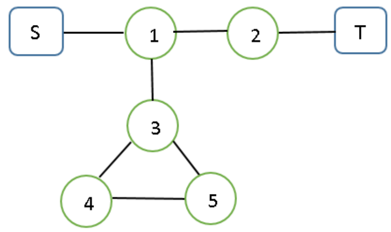

Figure 1: An example for non-complete graph

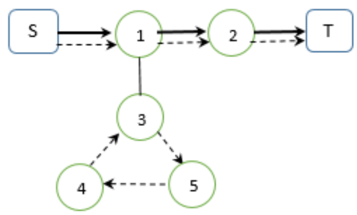

Figure 2: An example for disconnected subtour

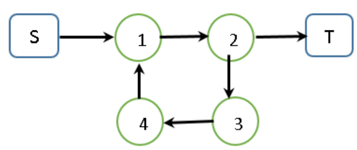

Figure 3: Arc $(1,2)$ must be passed twice

Table 1 summarizes notations of this paper.

Table 1: Summary of notations

U	set of vehicles
V	set of customers and source and terminal node
C	capacity of each vehicle in one rout
u_k	the k^{th} vehicle
S	source node
T	terminal node
i	index for customer
q_i	demand of customer i

The considered VRPNG consists of designing a feasible set of routes with minimum total length such that:

- each customer $i = 1, \ldots, n$ is serviced by exactly one vehicle.
- any route starts from node S and ends with node T.
- the maximum demand that a vehicle can serve is C;
- each customer might be visited several times by several vehicles.

Decision variables:

x_{ij}^k number of times vehicle u_k passes edge $(i, j) \in A$
a_{ij}^k equals 1 if vehicle u_k services node i while passing edge $(i, j) \in A$, and 0 otherwise
w_i^k equals 1 if vehicle u_k visits node i, and 0 otherwise
$y_{pq}^{k,i}$ equals 1 if vehicle u_k passes edge $(p, q) \in A$ in its route from node S to node i, and 0 otherwise

The following MIP is proposed:

$$\min \quad \sum_{k=1}^{K} \sum_{(i,j)\in A} c_{ij} x_{ij}^k, \tag{1}$$

$$s.t. \quad \sum_{j:(i,j)\in A} x_{ij}^k = \sum_{j:(j,i)\in A} x_{ji}^k, \quad \forall i \neq S, T, \forall k = 1, \ldots, K \tag{1a}$$

$$\sum_{j:(S,j)\in A} x_{Sj}^k = 1, \quad \forall k = 1, \ldots, K, \tag{1b}$$

$$\sum_{j:(j,T)\in A} x_{jT}^k = 1, \quad \forall k = 1, \ldots, K, \tag{1c}$$

$$\sum_{i=1}^{n} \sum_{j:(i,j)\in A} q_i a_{ij}^k \leq C, \quad \forall k = 1, \ldots, K \tag{1d}$$

$$a_{ij}^k \leq x_{ij}^k, \quad \forall k = 1, \ldots, K, \forall (i, j) \in A \tag{1e}$$

$$\sum_{k=1}^{K} \sum_{j:(i,j)\in A} a_{ij}^k = 1, \quad \forall i = 1, \ldots, n, \tag{1f}$$

$$w_i^k \leq M \sum_{j:(i,j)\in A} x_{ij}^k \quad \forall i \in V, \forall k = 1, \ldots, K, \tag{1g}$$

$$\sum_{j:(i,j)\in A} x_{ij}^k \leq M w_i^k \quad \forall i \in V, \forall k = 1, \ldots, K, \tag{1h}$$

$$\sum_{q:(S,q)\in A} y_{Sq}^{k,i} = w_i^k, \quad \forall i \neq S, \forall k = 1, \ldots, K, \tag{1i}$$

$$\sum_{q:(q,i)\in A} y_{qi}^{k,i} = w_i^k, \quad \forall i \neq S, \forall k = 1, \ldots, K, \tag{1j}$$

$$\sum_{q:(p,q)\in A} y_{pq}^{k,i} = \sum_{q:(q,p)\in A} y_{qp}^{k,i}, \quad \forall p \neq S, i,\ \forall k = 1, \ldots, K, \forall i \in V, \tag{1k}$$

$$y_{pq}^{k,i} \leq x_{pq}^k, \quad \forall (p, q) \in A, \forall k = 1, \ldots, K,\ \forall i \in V, \tag{1l}$$

$$x_{ij}^k \in Z^+, a_{ij}^k \in \{0, 1\}, \quad \forall (i, j) \in A,\ \forall k = 1, \ldots, K, \tag{1m}$$

$$y_{pq}^{k,i} \in \{0,1\}, \quad \forall (p,q) \in A, \; \forall k = 1,\ldots,K, \forall i \in V \tag{1n}$$

$$w_i^k \in \{0,1\}, \quad \forall i \in V, \; \forall k = 1,\ldots,K. \tag{1o}$$

The objective function (1) minimizes the total distance. Constraints (1a) assure that the number of arrivals to node i by vehicle u_k is equal to the number of departures from node i. In other words, if vehicle u_k arrive at any non-source and non-terminal node i, it must leave it. By constraints (1b) and (1c) any vehicle u_k leaves node S and enters node T exactly once respectively. Constraint (1d) is the capacity constraint. Serving node i by vehicle u_k is possible only if it passes edge $(i,j) \in A$ at least once, which is declared by constraint (1e). All customers are serviced exactly once by exactly one vehicle according to constraint (1f).

Based on the definition of variable w_i^k we must have:

$$w_i^k = \begin{cases} 1, & \sum_{j \in V} x_{ij}^k > 0, \\ 0 & \sum_{j \in V} x_{ij}^k = 0. \end{cases}$$

Constraints (1g) and (1h) define w_i^k. Constraint (1i)-(1l) indicate that if node i is visited by vehicle u_k, then there should be a path from node S to node i: constraints (1i) and (1j) guarantee leaving node S and entering node i, respectively, while constraint (1k) is the balance constraint. Constraint (1l) declares that arc (p,q) is contained in this path, only if vehicle u_k passes edge $(p,q) \in A$. Finally constraints (1m)-(1o) are domain constraints for decision variables of the model.
It is notable that the proposed model (1) involves $O(n^3K)$ decision variables and $O(n^3k)$ constraints, where n denotes the number of customers and K is the number of vehicles. We refer the reader to [4] for the classical model of VRP for complete graphs.

Example 1. Consider the graph $G = (V,A)$ in Figure 4.

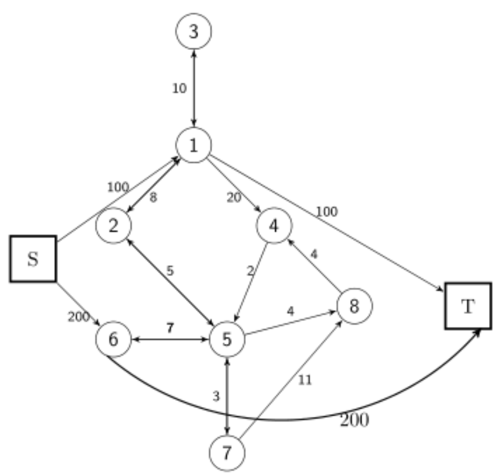

Figure 4: An example of a directed non-complete graph

Model (1) leads to the following solution, in which underlined nodes, are serviced in the rout:

$$\begin{aligned}
&route1: S-1-2-5-\underline{7}-5-\underline{8}-\underline{4}-5-2-1-T,\\
&route2: S-1-2-5-\underline{6}-\underline{5}-\underline{2}-1-T,\\
&route3: S-1-\underline{3}-\underline{1}-T.
\end{aligned}$$

The total cost is $Z^* = 702$. The classical model for VRP in [4] converts the graph to the complete graph by solving several shortest path problems. The following routes are achieved after solving VRP with the total cost of $Z^*_C = 706$:

$$\begin{aligned}
&route1: S-7-8-4-T;\\
&route2: S-2-6-5-T;\\
&route3: S-1-3-T;
\end{aligned}$$

Note that in the classical model, it is allowed to pass all arcs since the given graph is complete. For example, there is no edge between nodes S and 7 in the original graph, but the classical model solve several shortest path problems to achieve a complete graph, so there is an arc between S and 7 and its length is equal to $c_{S7} = c_{S1} + c_{12} + c_{25} + c_{57} = 116$. Other routes found by the classical model, have the same condition. The difference between Z^* and Z^*_C is arc $(7,8)$. Note that $c_{78} = 11$. In the first solution, although we have visited node 5 before, but we are allowed to pass the route$7-5-8$ with the cost of 7, instead of direct arc $(7,8)$ in the second solution, with the cost of 11.

Therefore converting the given graph to the complete one is not just time-consuming, but also eliminates the chance of passing an indirect but shorter path instead of direct arc in incident nodes, which may lead to visiting a node more that once.

3 Extended formulation

Disconnected subtour Elimination (DSE)constraints (1i)-(1l) are time consuming. In this section, we eliminate DSE constraints and provide a separation problem for the relaxed model:

$$\begin{aligned}
\min \quad & \sum_{k=1}^{K} \sum_{(i,j)\in \bar{E}} c_{ij} x^k_{ij}, \\
s.t. \quad & (1a)-(1h)\\
& (1m),(1o).
\end{aligned} \tag{2}$$

Solving model (2) leads to solutions $\bar{x}^k_{pq}$, $\bar{w}^k_i$ and $\bar{a}^k_{ij}$. First, note that for any node $i \in V$ and any vehicle u_k, DSE constraints provide a flow from node S to node i with $\bar{x}^k_{ij}$ as arc capacities. If vehicle u_k visits arc (p,q), then this arc is allowed to be included in the path from node S to node i. In other words, if $w^k_i \neq 0$, then there should be a flow from node S to node i. Therefore these constraints could be separated by a maximum flow problem for any node i and vehicle u_k. In the path from node S to node i by vehicle u_k, decision variables are:

$$\begin{aligned}
&f^{k,i}_{pq} = \text{amount of flow on arc } (p,q),\\
&flow(k,i) = \text{maximum flow from node } S \text{ to node } i.
\end{aligned}$$

For any i and vehicle k, we have the following maximum flow model with arc capacities $\bar{x}^k_{pq}$:

$$
\begin{aligned}
(SEP_{k,i}) \quad \max \quad & flow(k,i), \\
s.t. \quad & \sum_{q:(S,q)\in A} f^{k,i}_{Sq} = flow(k,i), \\
& \sum_{q:(q,i)\in A} f^{k,i}_{qi} = flow(k,i), \\
& \sum_{q:(p,q)\in A} f^{k,i}_{pq} = \sum_{q:(q,p)\in A} f^{k,i}_{qp}, \quad \forall p \neq S, i \\
& 0 \leq f^{k,i}_{pq} \leq \bar{x}^k_{pq}, \quad \forall (p,q) \in A,
\end{aligned}
\tag{3}
$$

If there exist node i and vehicle u_k for which $flow^*(k,i) = 0$, it means that there is no path from node S to node i in routeof vehicle u_k. So, node i is included in a disconnected subtour. As a result, constraints (1i)-(1l) for the mentioned k and i are added to the relaxed model (2) and it is solved again. The procedure is continued until $flow^*(k,i) > 0$ for any k,i.

Another approach is to add constraints of the separation problem to the relaxed model (2) to obtain an extended formulation, in a way that extended formulation is feasible if and only if $flow^*(k,i) > 0$ for any k,i. To do this, note that we do not need flow from node S to node i to be maximum, but we only need it to be positive, if vehicle u_k visits node i. So we can replace $flow(k,i)$ with w^k_i and add constraints to the relaxed model (2). If we consider variables $y^{k,i}_{pq}$ exactly the same as $f^{k,i}_{pq}$, then the extended formulation is turned to be equivalent with the original model (1).

The strength of model (1) becomes evident by the end of this section, as it is equivalent to its extended formulation. Therefore, the enhanced version of model (1), incorporating the valid inequalities introduced in Section 5, will be utilized.

4 Blocking procedure

In spite of disregarding complete graphs, VRPNG, is still an *NP*-hard problem. So, we provide a blocking procedure in order to reduce the size of the graph, in large size networks. In order to justify our procedure, consider waste collection problem, in which waste vehicles, with a limited capacity have to collect waste containers which are distributed according to a directed graph. Two types of aggregating nodes are defined as follows. For the first type blocks, consider a one-way street, with μ consecutive containers. It is clear that any vehicle arrives to the first container of this row, will continue its way up to the end of the row, unless the capacity is over. Then we can consider all nodes associated with this row as one node, and we call it *chain-block*. Any chain-block has a demand and a cost. Its demand is equal to the sum of the demands of nodes contained in the block, and its cost is equal to the distance from the first node to the last node of the block. The blocking procedure is done such that no chain-block has a demand more than C.

Now, for the second type blocks, consider a container at the end of deadlock alley, which is only accessible from one node. The last and the only container incident to the container in this alley may be considered as a block and we call it *access-block*. The same parameters, demand and cost are assigned to any access-block. The following subsection explains the detail about blocking procedure.

4.1 MIP for block network

A directed graph $G = (V,A)$ with arc distances c_{ij} for arc $(i,j) \in A$ is given. We define two types of blocks. Let $N^-(i) = \{j|(i,j) \in A\}$ and $N^+(i) = \{j|(j,i) \in A\}$ be out-neighbours and in-neighbours of node $i \in V$, respectively. Also $\delta^-(i) = |N^-(i)|$ and $\delta^+(i) = |N^+(i)|$ are out-degree and in-degree of node i, respectively.

Definition 1. *For graph $G = (V,A)$ with arc distances c_{ij}, subset of vertices $B = \{n_1,\ldots,n_p\} \subset V$, is called a block-chain of graph G, if $\delta^-(n_i) = \delta^+(n_i) = 1$, for $i = 1,\ldots,p$, and $N^-(n_i) = \{n_{i+1}\}$, for $i = 1,\ldots,p-1$. For chain-block B we have:*

$$demand(B) = \sum_{j=1}^{p} q_{n_j}, \quad cost(B) = \sum_{j=1}^{p-1} c(n_j,n_{j+1}).$$

Definition 2. *Subset of vertices $B = \{n_1,\ldots,n_p\} \subset V$, is called an access-block of graph G, if $N^-(n_1) = N^+(n_1) = \{n_2, j\}$, $N^-(n_i) = N^+(n_i) = \{n_{i-1},n_{i+1}\}$, for $i = 2,\ldots,p-1$, and $N^-(n_p) = N^+(n_p) = \{n_{p-1}\}$, where $j \in V \setminus B$. For access-block B we have:*

$$demand(B) = \sum_{j=1}^{p} q_{n_j}, \quad cost(B) = \sum_{j=1}^{p-1} [c(n_j,n_{j+1}) + c(n_{j+1},n_j)].$$

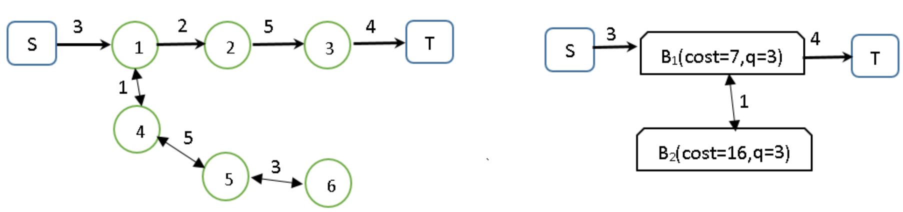

Figure 5: An example for the blocking procedure

Figure 5 shows blocking procedure in a simple graph.

Remark 1. *When we enter a chain-block, first we visit its first node and when we exit a chain-block, its last node is our last visit. However, when we enter an access-chain, first and last node we visit, is its first node. According to this, we update arc distances in an aggregated graph. The first node and the last visited node of block B is shown by $B(first)$ and $B(last)$, respectively.*

After building chain-blocks and access-blocks of graph G, we have an aggregated graph $G^B = (V^B, A^B)$. Note that if there exists a node $i \in V$ that does not belong to any block, then for simplicity it is assumed as a single chain-block with zero cost. Arc distances c^B_{ij} are built as follows:

$$
\begin{aligned}
V^B =& \{B_i | B_i \text{ is a chain-block or an access-block in graph } G\},\\
A^B =& \{(B_i, B_j) | (B_i(last), B_j(first)) \in A, B_i \text{ is a chain-block}\} \cup\\
& \{(B_i, B_j) | (B_i(first), B_j(first)) \in A, B_i \text{ is an access-block}\},\\
c^B_{B_i,B_j} =& \begin{cases} c(B_i(last), B_j(first)), & B_i(last), B_j(first) \in A\&, B_i \text{ is a chain-block}\\ c(B_i(first), B_j(first)), & B_i(first), B_j(first) \in A\&, B_i \text{ is an access-block}\end{cases}
\end{aligned}
$$

Up to know, we have built an aggregated graph $G^B = (N^B, E^B)$, in which any block node $B_i \in V$ has the cost of $cost(B_i)$ and demand of $q(B_i)$. Moreover, arc distance $c^B(B_i, B_j)$ is assigned to arc (B_i, B_j). In order to solve the problem on the reduced aggregated graph, first we have to modify the model for graphs, in which nodes have costs. It only suffices to change the objective function (1) to the form:

$$
\min \quad \sum_{k=1}^{K} \sum_{(i,j)\in A^B} c^B_{ij} x^k_{ij} + \sum_{k=1}^{K} \sum_{B_i \in V^B} cost(i) w^k_i \tag{4}
$$

5 Valid inequalities

In this section we provide two types of valid inequalities to strengthen our MIP model. First we introduce two subsets of arcs as follow:

$$
\begin{aligned}
A^- &= \{(i,j) \in A | \delta^-(i) = 1\},\\
A^+ &= \{(i,j) \in A | \delta^+(j) = 1\}.
\end{aligned}
$$

It is clear that if $(i,j) \in A^-$, then serving customer $i \in V$ is only possible by passing arc (i,j). In other words, one of the K vehicles has to pass arc (i,j) and service node i. This can be shown by the following inequality:

$$
\sum_{k=1}^{K} a^k_{ij} = 1, \quad \forall (i,j) \in A^-. \tag{5}
$$

On the other hand, if $(i,j) \in A^+$, and if vehicle u_k is chosen to service node j, then it has to pass arc (i,j), or serving node j is only possible by passing arc (i,j). In other words

$$
\sum_{p \in V} a^k_{jp} = 1 \implies x^k_{ij} \geq 1, \quad \forall (i,j) \in A^+,
$$

which is done by the following inequality:

$$
x^k_{ij} \geq \sum_{k=1}^{K} a^k_{ij}, \quad \forall (i,j) \in A^+. \tag{6}
$$

Adding valid inequalities (5) and (6) to model (1) or (4) provide a stronger formulation for VRPNG.

Table 2: Results for library networks

Network Parameters		Optimal solution (First incumbent)			Time(min:sec)			Duality gap		
Name	$(\lvert V\rvert,\lvert A\rvert)$	P.	St. P.	Bl. P.	P.	St. P.	Bl. P.	P.	St. m.	Bl. P.
ta1	(24,55)	967	967	967	1:1	1:06	1:06	0	0	0
giul39	(39,170)	1058	1058	1058	2:46	2:32	2:32	0	0	0
germany50	(50,88)	7506	6430	6430	15:54	12:16	12:15	3.1	2.82	2.82
zib54	(54,80)	8706	7542	7412	42:41	25:15	22:18	24.15	18:16	17:08
ta2	(65,108)	10256	9861	6541	117:05	35:13	33:40	30.13	16.15	6.03

6 Numerical results

In this section, we implemented our model on two types of networks. The first type contains library networks and the second one contains a network obtained in a real case study. We implemented the proposed model in Gams 25.1.2 on a computer with Intel(R) Core(TM) i7-5500U 2.4 GHz processor and 8 GB RAM. CPLEX 12.8 is chosen as the MIP solver.

6.1 Library networks

The network instances in this section are obtained from SNDlib.zib.de, which is a library of test instances for network optimization. The results are outlined in Table 2. Column entitled "P." shows the results obtained by solving model (1), column entitled "St. P." shows the results related to the model (1) strengthen with valid inequalities (5) and (6). Finally the column entitled "Bl. P." is related to solving the strengthened model (1) on the reduced network after blocking procedure.

Table 2 shows that as size of the network grows gradually, the proposed model with valid inequalities and blocking procedure outgo the original model (1). Therefore, larger the sample size, more effective strengthening and compacting the network. This is more clear in our case study in the next subsection.

6.2 Behshahr case description

The waste collection system in Behshahr is a simple VRPNG. Behshahr is a city in Mazandaran province, Iran. It is located at the foot of the Alborz mountains. There are 800 waste containers that are clustered into 5 regions for 5 identical vehicles with capacity of 43 containers for each. The clustering procedure has been done by Behshahr Municipality according to its predetermined regions.
Any container starts its first route from parking and ends it to a depot, but other routes starts and ends in depot. Parking and depot are close to each other and both of them are in the countryside, that can be assumed as one node. Therefore we assumed that every route starts from source (parking) and ends to terminal (depot).

In this problem, all containers are assumed to be full and ready for evacuation. So, it is clear that the demand of node (container) i is one unit for $i = 1,\dots,n$ and zero for source and terminal nodes. Any region contains about 160 waste containers. Any vehicle is responsible for its own region and have to traverse 4 routes from S to T to collect all containers.

Therefore, the problem of designing 4 routes for one vehicle in any region is VRPNG for 4 vehicles $\{u_1, \dots, u_4\}$ with 43 units of capacity. Here we have $G = (V,A)$, where $V = \{S, 1, \dots, 160, T\}$. In order to obtain set of edges A, we suppose $(i, j) \in A$, if and only if there is no other container in the shortest path from container located at node i to container at j and c_{ij} shows the shortest path from node i to j.

After preparing the graph $G = (V,A)$ in the text file, four approaches were applied to solve the problem:

1. The first approach: Solve model (1) for graph G;
2. The second approach: Build an aggregated graph and solve model (4);
3. The third approach: Strengthen formulation (4) with valid inequalities (5) and (6) and solve it for the aggregated graph;
4. The fourth approach: Solve several shortest path problems on the aggregated graph to achieve the complete graph and solve the usual model of VRP with node costs (see [4]).

Table 3 shows the results for the first region with 160 nodes and 406 edges.

Table 3: Results of 4 approaches for the first region

Approach	N	E	Incumbent value	Time hour:min	Duality gap	Reduction (Km)
1	160	406	4478	24:10	9.70%	5.270
2	149	393	4374	24:10	9.11%	6.111
3	149	393	4329	24:10	7.32%	6.494
4	149	11026	no integer solution found (Out of memory)	4:27	-	-

The driver responsible for this region was driving four routes with the total length of $Z = 5098$ in map scale. The third column labeled Incumbent value, is the value of the solution obtained after time limitation stopped the program. The last column labelled Reduction, shows the reduction in objective value comparing to the current solution by driver in Kilo-meter. It can be seen that the 3 first approaches based on non-complete graphs are successful in finding integer solution better than current one in a day, but the classical model for VRP for complete graphs ran out of memory in about 4 hours without finding even an integer solution. Similar results were obtained in other regions.

Table 4 shows the results for the second region with 158 nodes and 395 edges. The driver responsible for the second region was driving four routes with the total length of $Z = 5000$.

Table 4: Results of 4 approaches for the second region

Approach	N	E	Incumbent value	Time hour:min	Duality gap	Reduction (Km)
1	158	395	4462	24:10	9.68%	4.573
2	142	380	4351	24:10	9.42%	5.516
3	142	380	4239	24:10	7.81%	6.468
4	142	10011	no integer solution found (Out of memory)	4:45	-	-

Table 5 shows the results for the third region with 161 nodes and 401 edges. The driver responsible for the third region was driving four routes with the total length of $Z = 4985$.

Table 5: Results of 4 approaches for the third region

Approach	N	E	Incumbent value	Time hour:min	Duality gap	Reduction (Km)
1	161	401	4531	24:10	8.86%	3.859
2	148	395	4500	24:10	8.23%	4.122
3	148	395	4384	24:10	6.81%	5.108
4	148	10878	no integer solution found (Out of memory)	4:35	-	-

Table 6 shows the results for the fourth region with 162 nodes and 404 edges. The driver responsible for this region was driving four routes with the total length of $Z = 5001$.

Table 6: Results of 4 approaches for the fourth region

Approach	N	E	Incumbent value	Time hour:min	Duality gap	Reduction (Km)
1	162	404	4426	24:10	9.13%	4.887
2	148	390	4306	24:10	8.56%	5.907
3	148	390	4201	24:10	6.73%	6.800
4	148	10878	no integer solution found (Out of memory)	4:15	-	-

Table 7 shows the results for the fifth region with 159 nodes and 400 edges. The driver responsible for this region was driving four route s with the total length of $Z = 5045$.

Table 7: Results of 4 approaches for the fifth region

Approach	N	E	Incumbent value	Time hour:min	Duality gap	Reduction (Km)
1	159	400	4651	24:10	8.23%	3.349
2	145	390	4425	24:10	8.03%	5.270
3	145	390	4382	24:10	6.61%	5.635
4	145	10440	no integer solution found (Out of memory)	5:11	-	-

Finally it can be seen that modelling the VRP for non-complete graphs is really affordable compared to classical model for complete graphs. Moreover reducing the graph size by blocking procedure and strengthening the formulation by valid inequalities make it applicable for even large size graphs in real case studies.

7 Conclusion

This paper studied vehicle routing problem for directed non-complete graphs (VRPNG). Classical models for VRP may be infeasible for non-complete graphs, since subtours are unavoidable for them. We proposed a mixed integer linear programming model for VRPNG, in which subtours are allowed but all nodes should be connected to the source node. This is guaranteed by disconnected subtour elimination (DSE) constraints. Separating DSE constraints and providing an extended formulation based on the separation problem results in our original proposed model, which is a reason for strength of our model.
We also proposed a blocking procedure to compact our directed graph and reduce its size. Finally, we provided two types of valid inequalities. In numerical results it was shown that some fractional solution are cut off by these cuts and duality gap is reduced.

Acknowledgement

Special thanks are due to Mohammad Javan and Vahid Ghorbani for their outstanding assistance with the preparation of the graph of all regions and Dr. Kamandi for his valuable comments.

References

[1] F. Alesiani, G. Ermis, K. Gkiotsalitis, *Constrained clustering for the capacitated vehicle routing problem*, In 101st Transportation Research Board (TRB) Annual Meeting 2022.

[2] J. H. Bhrmann, F. Bruwer, *K-medoid petal-shaped clustering for the capacitated vehicle routing problem*, S. Afr. J. Ind. Eng. **32** (2021) 33–41.

[3] E. Choi, D.W. Tcha, *A column generation approach to the heterogeneous fleet vehicle routing problem*, Comput. Oper. Res. **34** (2007) 2080–2095.

[4] N. Christofides, A. Mingozzi, P. Toth, *The Vehicle Routing Problem*, University of Waterloo, 1979, 315–338.

[5] L. Costa, C. Contardo, G. Desaulniers, *Exact branch-price-and-cut algorithms for vehicle routing*, Transp. Sci. **53** (2019) 946–985.

[6] R. Dantzig, J. Ramzer, *The truck dispatching problem*, Manag. Sci. **6** (1959) 81–91.

[7] A. Gunawan, AT. Widjaja, P. Vansteenwegen, F.Y. Vincent, *A matheuristic algorithm for the vehicle routing problem with cross-docking*, Appl. Soft Comput. **103** (2021) 107163.

[8] D. Jayarathna, *Survey on thirty years of vehicle routing problems: Mathematical models, solution methods, and real-life applications*, Int. J. Res. Sci. Innov. **11** (2024) 435–449.

[9] J.K. Lenstra, A.H.G. Rinnooy Kan, *Complexity of vehicle routing and scheduling problems*, Networks **11** (1981) 221–227.

[10] F. Liu, C. Lu, L. Gui, Q. Zhang, X. Tong, M. Yuan, *Heuristics for vehicle routing problem: A survey and recent advances*, 2023, arXiv:2303.04147.

[11] K. E., Medarhri, R. Zine, *A survey of the vehicle routing problem and its variants: formulations and solutions*, Math. Model. Comput. **11** (2024) 333–343.

[12] P. Munari, T. Dollevoet, R. Spliet, *A generalized formulation for vehicle routing problems*, 2017, arXiv:1606.01935v2.

[13] E. Queiroga, R. Sadykov, E. Uchoa, RP. da Ptria, *A modern POPMUSIC matheuristic for the capacitated vehicle routing problem*, ech. rep. Caderno do LOGIS, Universidade Federal Fluminense, 2020.

[14] M. Sadat Emami Taba, *Solving Travelling Salesman Problem With a non-complete Graph*, Master's Thesis, Waterloo University, 2009.

[15] C. Shi, T. Li, Y. Bai, F. Zhao, *A heuristics-based parthenogenetic algorithm for the VRP with potential demands and time windows*, Sci. Programming **2016** (1016) 8461857.

[16] A. Tuzemen, C. Yildiz, *A Solution proposal to vehicle routing problem with integer linear programming: A distributor company sample*, Int. J. Contemporary Economics and Administrative Sci., **9** (2019) 46–78.