

Valid Inequalities for the Vehicle Routing Problem with Drones

Faiz Hamid¹ and Yogesh K. Agarwal²

¹Department of Management Sciences, Indian Institute of Technology Kanpur, Kanpur 208016, India, ✉ fhamid@iitk.ac.in

²Operations and Decision Sciences Area, Jaipuria Institute of Management, Lucknow, Lucknow 226010, India, ✉ yogesh.agarwal@jaipuria.ac.in

1 Introduction

The Vehicle Routing Problem with Drones (VRPD) extends the classical Capacitated Vehicle Routing Problem by allowing drones to cooperate with trucks for last-mile delivery. Drones are well suited for short-range, low-weight deliveries and can avoid road congestion, while trucks provide higher capacity and longer-range transportation. Motivated by emerging applications, this paper studies a VRPD variant in which drones operate in tandem with trucks and may also be dispatched directly from the depot, in which case they must return immediately after service.

The problem is modeled on a complete directed graph $G = (N, A)$, where $N = \{0\} \cup C$ denotes the depot and customer set. Each customer $i \in C$ has demand q_i . Vehicles have capacity Q^V , while drones have capacity Q^D and endurance limit e . Let $C^D \subseteq C$ and $C^V \subseteq C \setminus C^D$ denote the sets of drone-eligible and drone-ineligible customers, respectively. Travel costs and times for vehicles and drones on arc (i, j) are denoted by (c_{ij}^V, t_{ij}^V) and (c_{ij}^D, t_{ij}^D) , with $t_{ij}^V \geq t_{ij}^D$. At most μ drones may operate in tandem with a vehicle.

Binary decision variable x_{ij} equals 1 if a vehicle travels directly from node i to node j , and 0 otherwise. Variables z_{ij} and w_{ij} equal 1 if a drone travels from i to j in loaded and empty states, respectively. Binary variable y_i equals 1 if customer i is served by a drone. Continuous variable u_i denotes the vehicle load immediately before visiting customer i . The problem is formulated mathematically as follows.

$$\text{Minimize } \sum_{(i,j) \in A} c_{ij}^V x_{ij} + \sum_{(i,j) \in A} c_{ij}^D (z_{ij} + w_{ij}) \quad (1)$$

$$\text{subject to } \sum_{j \in N} x_{ji} - \sum_{j \in N} x_{ij} = 0 \quad \forall i \in C \quad (2)$$

$$\sum_{j \in N} z_{ji} - \sum_{j \in N} w_{ij} = 0 \quad \forall i \in C \quad (3)$$

$$\sum_{j \in N} x_{ji} + \sum_{j \in N} z_{ji} = 1 \quad \forall i \in C \quad (4)$$

$$t_{ij}^D z_{ij} + t_{jk}^D w_{jk} \leq e \quad \forall j \in C, i, k \in N \quad (5)$$

$$z_{ij} + w_{jk} \leq 1 + x_{ik} \quad \forall i, j, k \in C \quad (6)$$

$$w_{i0} \geq z_{0i} \quad \forall i \in C \quad (7)$$

$$u_i \geq u_j + \sum_{k \in C} q_k z_{ik} + q_i - Q^V (1 - x_{ij}) \quad \forall i, j \in C \quad (8)$$

$$u_i \geq \sum_{k \in C} q_k z_{ik} + q_i - Q^V (1 - x_{i0}) \quad \forall i \in C \quad (9)$$

$$u_i \leq Q^V \quad \forall i \in C \quad (10)$$

$$\sum_{j \in C} z_{ij} \leq \mu \sum_{j \in N} x_{ji} \quad \forall i \in C \quad (11)$$

$$\sum_{j \in N} z_{ji} = y_i \quad \forall i \in C \quad (12)$$

$$y_i = 0 \quad \forall i \in C^V \quad (13)$$

$$x_{ij}, y_i, z_{ij} \in \{0, 1\} \quad \forall (i, j) \in A \quad (14)$$

$$u_i \geq q_i \quad \forall i \in C \quad (15)$$

The objective function (1) minimizes the total operational cost of trucks and drones. Constraints (2) enforce vehicle flow conservation at customer nodes. Constraints (3) ensure that each loaded drone arriving at a customer departs empty. Constraints (4) guarantee that each customer is served exactly once, either by a truck or a drone. Constraints (5) impose the drone endurance limit on launch–service–rejoin operations. Constraints (6) restrict drone rejoining to the same or the subsequent vehicle stop. Constraints (7) require drones launched from the depot to return directly to the depot. Constraints (8)–(9) are MTZ-type constraints that enforce vehicle capacity feasibility and eliminate subtours. Constraints (10) bound the vehicle load by its capacity. Constraints (11) limit the number of simultaneously deployed drones by the available fleet size. Constraints (12) link customer assignment to incoming drone arcs. Constraints (13) prohibit drone delivery for drone-ineligible customers. Constraints (14)–(15) define variable domains.

Computational studies show that the LP relaxation of standard VRPD formulations is weak, and as a result most large-scale instances are solved using heuristics [1], while exact methods remain limited to small instances [2]; this motivates the identification of valid inequalities to significantly strengthen the MILP formulation.

2 Identification of Valid Inequalities

To strengthen the linear relaxation of the proposed VRPD formulation, we identify several classes of valid inequalities adapted from the VRP literature.

Let $S \subseteq C$ and $\bar{S} = C \setminus S$. For any $t \in S$, customer t must either be served by a drone or at least one vehicle must enter S . This yields the following generalized subtour elimination (GSEC) inequality:

$$\sum_{j \in \bar{S}} \sum_{i \in S} x_{ji} + y_t \geq 1 \quad \forall S \subseteq C, t \in S. \quad (16)$$

Let $S \subseteq C^V$. Capacity considerations impose a lower bound on the number of vehicles entering S leading to the following rounded capacity inequality (RCI):

$$\sum_{j \in \bar{S}} \sum_{i \in S} x_{ji} \geq \left\lceil \frac{\sum_{i \in S} q_i}{Q^V} \right\rceil. \quad (17)$$

We also introduce the following inequality that restricts infeasible combinations of vehicle and drone movements between customer pairs:

$$2x_{ij} + 2x_{ji} + z_{ij} \leq 2 - y_i - y_j \quad \forall i, j \in C. \quad (18)$$

This inequality enforces consistency between service modes: a drone arc is allowed only if at least one of the two customers is served by a vehicle, while vehicle arcs are permitted only between customers not simultaneously assigned to drone service.

Additional classes of valid inequalities (such as endurance-based RCIs, symmetry breaking, etc.) are derived to further strengthen the formulation, but are omitted here due to space limitations.

3 Computational Study

The proposed MILP formulation and valid inequalities were implemented in C++ and solved using the Gurobi 13.0.0 solver. Computational experiments were conducted on 25 randomly generated instances with 30 customers each. All valid inequalities were added only at the root node; once no further violated inequalities were detected, the resulting model was solved using Gurobi's default branch-and-bound settings with a time limit of 2 hours.

We compare two configurations: (i) the formulation strengthened with previously known VRP cuts (Prev), and (ii) the formulation augmented with the proposed VRPD-specific valid inequalities (User). Performance is evaluated in terms of root-node lower bounds, total CPU time, and final optimality gaps.

Table 1 summarizes the results. The proposed inequalities substantially improve the quality of the LP relaxation, increasing the average root-node lower bound from 85.90% to 94.36%. This strengthening translates into a dramatic reduction in solution time and optimality gaps: all 25 instances are solved to optimality with an average CPU time of 482 seconds. In contrast, using previously known cuts, only 7 instances are solved to optimality within the time limit, with an average optimality gap of 4.23%.

Table 1: Computational results on 30-customer VRPD instances

Inst.	Z_{IP}^*	Lower Bound (%)			Time (s)		Gap (%)	
		LB0	Prev	User	Prev	User	Prev	User
A1	5866	29.46	82.60	92.71	7201	337	7.64	0.00
A2	5555	32.38	90.92	94.57	1281	258	0.00	0.00
A3	6344	28.65	86.55	93.42	7200	105	2.43	0.00
A4	5402	36.58	87.52	92.10	7200	407	5.46	0.00
A5	6493	28.91	84.80	96.20	7201	154	5.65	0.00
A6	6014	37.04	84.44	92.45	7200	146	8.63	0.00
A7	5358	37.64	91.72	94.85	1272	449	0.00	0.00
A8	5204	41.96	89.03	93.29	1004	77	0.00	0.00
A9	6507	21.56	86.35	96.52	7200	204	3.32	0.00
A10	5356	34.06	87.73	94.43	7039	120	0.00	0.00
A11	5364	33.76	87.55	96.06	3914	141	0.00	0.00
A12	5396	37.19	82.54	93.88	7201	2950	5.52	0.00
A13	5146	37.69	87.90	97.19	7200	51	2.27	0.00
A14	6454	24.84	82.27	93.85	7201	112	3.08	0.00
A15	5294	30.22	86.27	91.86	7201	836	8.01	0.00
A16	4978	41.95	84.72	92.66	7201	235	5.12	0.00
A17	4589	35.76	80.73	93.10	7200	155	6.56	0.00
A18	7146	30.86	84.81	93.97	7201	1716	7.64	0.00
A19	4986	34.56	79.96	92.53	7200	349	15.70	0.00
A20	5322	30.08	81.93	95.76	7200	2143	10.05	0.00
A21	4290	32.96	81.44	96.11	1841	43	0.00	0.00
A22	5316	32.13	89.48	95.66	7200	205	3.69	0.00
A23	4694	37.19	91.82	95.54	7200	401	1.70	0.00
A24	5418	43.76	88.44	94.51	431	70	0.00	0.00
A25	5208	37.56	86.04	95.68	7201	386	3.19	0.00
Avg.		33.95	85.90	94.36	5855.6	482.0	4.23	0.00
# Opt.							7	25

4 Conclusion and Future Work

This study presents a strengthened MILP formulation for the VRPD by incorporating several classes of valid inequalities that significantly improve the linear relaxation making the approach suitable for exact solution methods. Future work will focus on the systematic separation of these inequalities, integration within a branch-and-cut framework, and computational evaluation on larger benchmark instances. Extensions to multiple vehicles, heterogeneous drones, and endurance-dependent synchronization constraints also constitute promising research directions.

5 References

1. Schermer, D., Moeini, M., & Wendt, O. (2019). A matheuristic for the vehicle routing problem with drones and its variants. *Transportation Research Part C: Emerging Technologies*, 106, 166-204.
2. Wang, Z., & Sheu, J. B. (2019). Vehicle routing problem with drones. *Transportation research part B: methodological*, 122, 350-364.