The parallel drone scheduling problem with multiple drones and vehicles

Raïssa G. Mbiadou Saleu, Laurent Deroussi, Dominique Feillet, Nathalie Grangeon, Alain Quilliot

European Journal of Operational Research (2021)

Drones for last-mile delivery

Fig. 4. UPS HorseFly.

Fig. 3. DHL Parcelcopter.

Combined with trucks

- Flying Sidekick Traveling Salesman Problem (FSTSP)
    - drone travels with truck
    - with synchronization
- Parallel Drone Scheduling Traveling Salesman Problem (PDSTSP)
    - drone departs from depot
    - no synchronization

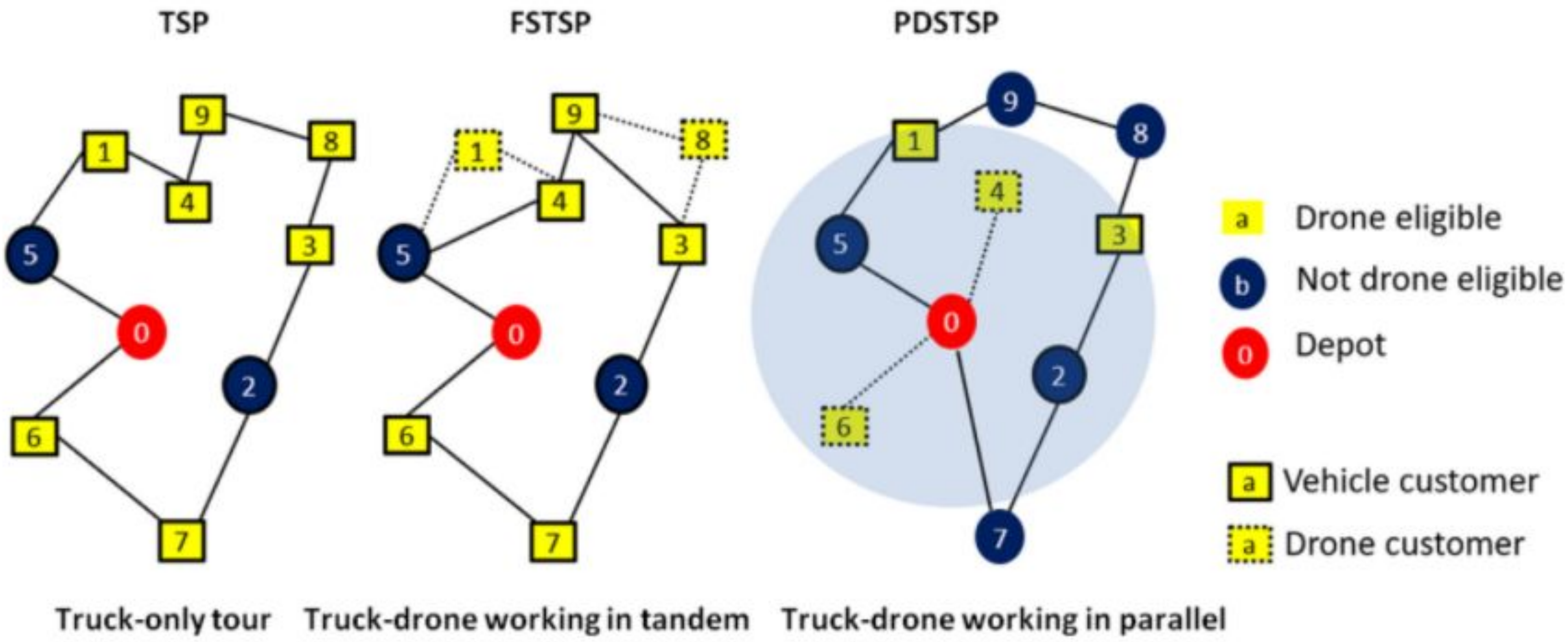


Fig. 5. Illustration of FSTSP and PDSTSP vs truck-only delivery.

This paper

- Parallel Drone Scheduling Travelling Salesman Problem (PDSTSP)
- one depot, multiple trucks and multiple drones
- Mixed Integer Linear Programming Formulation (MILP)
- hybrid metaheuristic

Problem statement

$G = (N \cup \{0\}, A)$

N: customers

N_d: drone-eligible customers

0: depot

K: trucks

M: drones

$t_{i,j}$: travel time for a truck

$\hat{t}_i$: travel and delivery time for a drone

Objective

- Minimize delivery completion time
    - all deliveries completed
    - all trucks and drones at depot

MILP formulation

- decision variables

z_i: customer i visited by vehicle (otherwise drone)
$x_{i,j,k}$: arc (i, j) belongs to vehicle tour k
$w_{i,j}$: arc belongs to a vehicle tour
$y_{i,m}$: customer i assigned to drone m
T: completion time

MILP formulation

- completion time at least as big as greatest truck travel time

$$T \geq \sum_{(i,j)\in A} t_{ij} x_{ijk} \quad (1 \leq k \leq K)$$

MILP formulation

- completion time at least as big as greatest drone travel time

$$T \geq \sum_{i \in N_d} \widehat{t_i} y_{im} \quad (1 \leq m \leq M)$$

MILP formulation

- serve not drone-eligible customers by truck

$$z_i = 1 \quad (i \in N \setminus N_d)$$

MILP formulation

- customer is served by either truck or drone

$$\sum_{1 \le m \le M} y_{im} = 1 - z_i \quad (i \in N_d)$$

MILP formulation

- if customer served by truck, it is part of a truck tour

$$w_{ij} = \sum_{1 \le k \le K} x_{ijk} \quad ((i, j) \in A)$$

$$\sum_{(i,j) \in A} w_{ij} = z_i \quad (i \in N)$$

MILP formulation

- each truck leaves the depot at most once

$$\sum_{(0,j)\in A} x_{0jk} \leq 1 \quad (1 \leq k \leq K)$$

MILP formulation

- flow conservation for truck tours

$$\sum_{(i,j)\in A} x_{ijk} = \sum_{(j,i)\in A} x_{jik} \quad (i \in N,\ 1 \le k \le K)$$

MILP formulation

- Subtour Elimination Constraint

$$\sum_{j \in S} \sum_{l \in N \cup \{0\} \setminus S} w_{jl} \geq z_i \quad (S \subseteq N, S \neq \emptyset, i \in S)$$

MILP formulation

$$z_i \in \{0, 1\} \quad (i \in N)$$

$$x_{ijk} \in \{0, 1\} \quad ((i, j) \in A, \; 1 \leq k \leq K)$$

$$w_{ij} \in \{0, 1\} \quad ((i, j) \in A)$$

$$y_{im} \in \{0, 1\} \quad (i \in N_d, 1 \leq m \leq M)$$

$$T \geq 0$$

Hybrid metaheuristic

- extension of a procedure by Mbiadou Saleu et al. (2018)
    - PDSTSP
    - one truck
    - one depot

Hybrid metaheuristic

- extension of a procedure by Mbiadou Saleu et al. (2018)
    - TSP tour τ visiting all customers
    - decompose τ
        - subsequence τ_{vehicle}
        - subset π_{drones}
        - dynamic programming with labeling

Hybrid metaheuristic

- extension of a procedure by Mbiadou Saleu et al. (2018)
    - TSP tour τ visiting all customers
    - decompose τ
    - re-optimize $\tau_{vehicle}$ with Lin-Kernighan heuristic
    - assign customers from π_{drones} to drones
        - Parallel Machine Scheduling (PMS)
        - greedy heuristic

Hybrid metaheuristic

- extension of a procedure by Mbiadou Saleu et al. (2018)
  - TSP tour τ visiting all customers
  - decompose τ
  - re-optimize $\tau_{vehicle}$ with Lin-Kernighan heuristic
  - assign customers from π_{drones} to drones
  - construct new τ for next iteration

Hybrid metaheuristic

- this paper
    - multiple vehicle tours needed
    - solution S:
        - K customer sequences
        - M customer sets

Hybrid metaheuristic

- algorithm
  - Initialization
  - Decoding
  - Route re-optimization
  - Drone assignment
  - Local search
  - Improve
  - Construct new giant tour

Hybrid metaheuristic

- Initialization
  - giant TSP tour τ
  - nearest-neighbour construction procedure
  - all visitors visited by a single vehicle

Hybrid metaheuristic

- Decoding
    - decompose τ
    - K subsequences
    - one set π_{drones}

Hybrid metaheuristic

- Route re-optimization
    - re-optimize vehicle routes
    - TSP Lin-Kernighan heuristic
    - Helsgaun’s implementation (Helsgaun, 2000)

Hybrid metaheuristic

- Drone assignment
    - assign customers to drones
    - Parallel Machine Scheduling (PMS)
    - longest processing time heuristic (Pinedo & Hadavi, 1992)

Hybrid metaheuristic

- Local search
  - improve current solution
- Update best solution
  - smaller completion time
  - same completion time and smaller total travel time

Hybrid metaheuristic

- Construct new giant tour
    - concatenate vehicle tours in random order
    - randomly insert customers assigned to drones
    - optimize result with 2-opt
- continue iterated local search (ILS)
- stop when time limit is reached

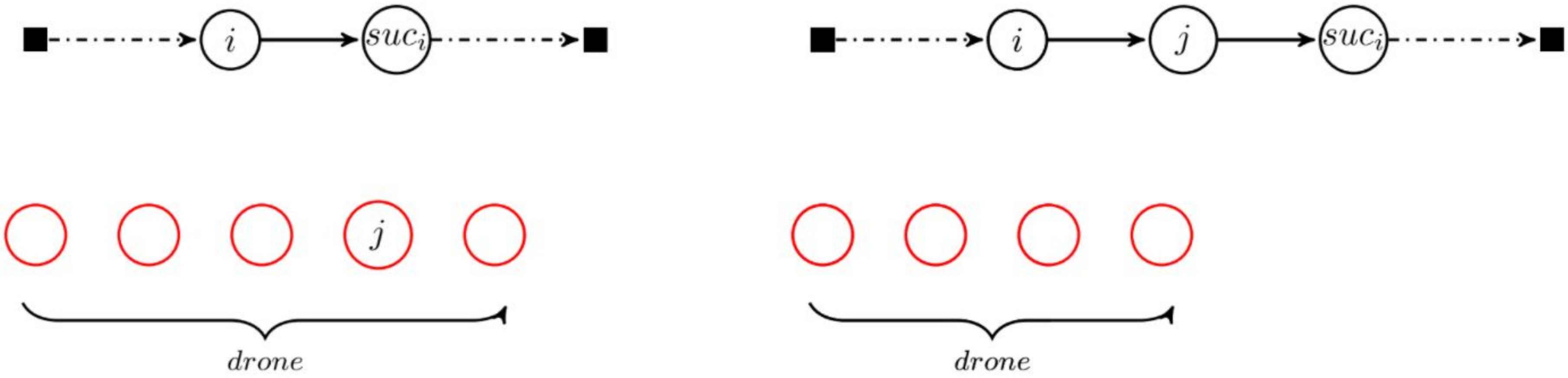
Local search

- improve current solution
- moves:
    - Transfer
    - Exchange move drone - vehicle / vehicle - vehicle
    - Relocate
    - Cross move
- apply moves until local optimum

Transfer move

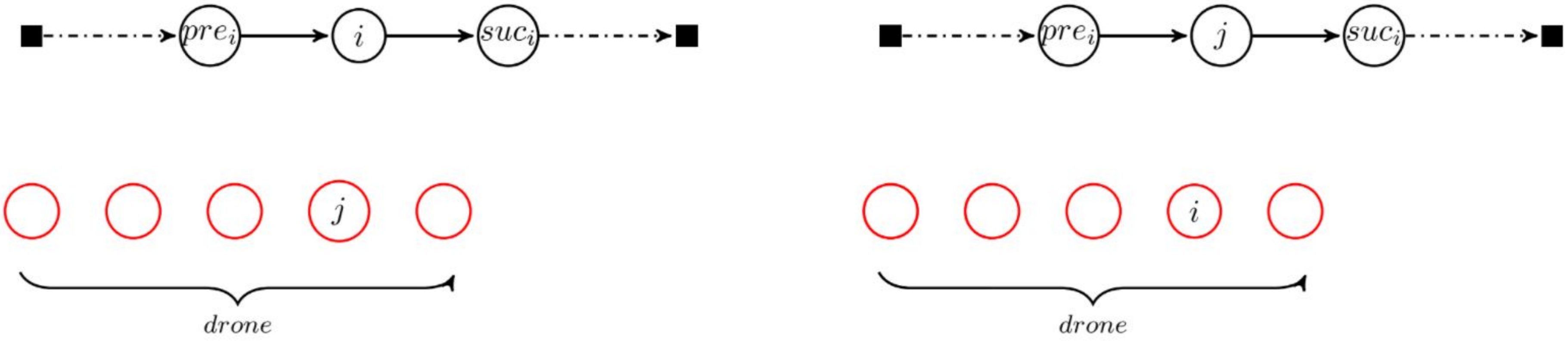
(a) transfer from a vehicle to a drone

Transfer move



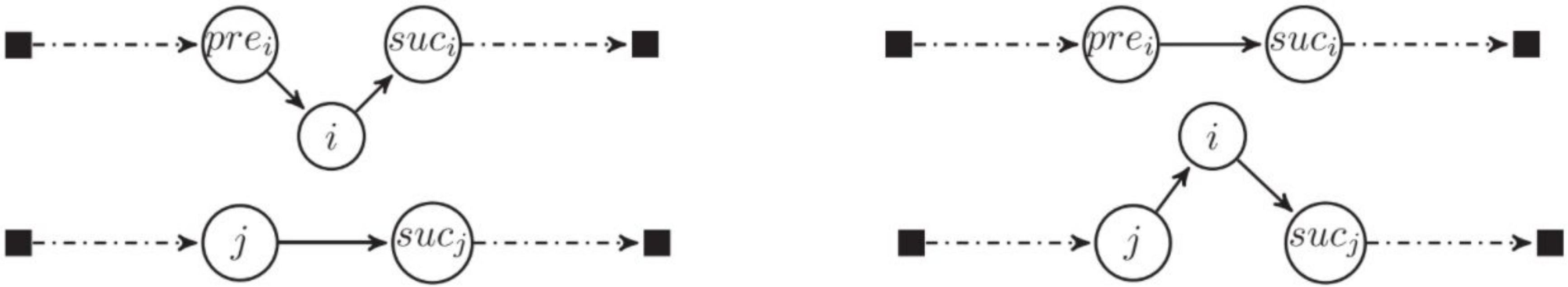
(b) transfer from a drone to a vehicle

Exchange move drone - vehicle



(c) exchange move drone-veh

Relocate move

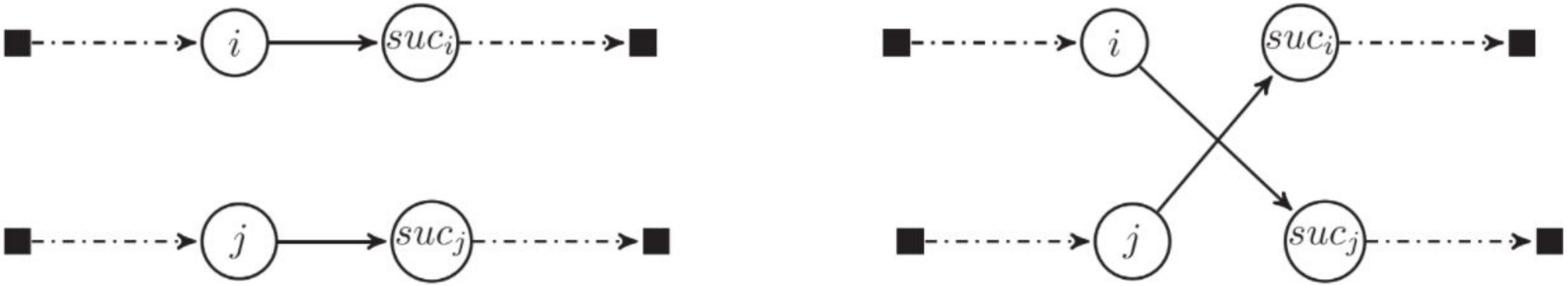


(a) relocate move

Exchange move vehicle - vehicle

(b) exchange move veh-veh

Cross move



(c) cross move

Decoding

- split procedure:
  - extracts K vehicle tours and set of customers served by drones
  - introduce acyclic directed graph
  - solve multi-criteria shortest path problem by dynamic programming (Climaco & Martins, 1982)
  - assign labels to nodes

Decoding

- split procedure
- upper bounds
- lower bounds
- bounding mechanism for pruning labels

Experiments

- CVRPLIB dataset
    - 20 instances
    - depot near barycenter of all customers
    - between 50 and 199 customers

Results

- MILP
  - branch-and-cut algorithm
  - up to 100 customers
  - time limit of 3 hours
  - no optimal solutions found

Results

- Hybrid metaheuristics (HM)
- different variants:
  - HMb
  - MS
  - HM(LL), HMb(LL), MS(LL)
  - HM(UB), HMb(UB), MS(UB)
- time limit of 1000 seconds

Hybrid metaheuristics (HM) variants

- HMB
    - modified reconstruction of the giant tour
    - parameter X
    - concatenate first X vehicle tours of best solution in random order
    - best insertion for remaining customers

Hybrid metaheuristics (HM) variants

- HMB
  - update X:
  - current solution better than best solution:
    - set X = K
  - otherwise:
    - set X = max (0, X - 1)

Hybrid metaheuristics (HM) variants

- MS
  - generate giant tour with randomized nearest-neighbour heuristic
  - randomly choose one of three nearest neighbours
  - standard multi-start (instead of ILS)

Hybrid metaheuristics (HM) variants

- HM(LL), HMb(LL), MS(LL)
    - labels in the decoding step are limited
    - less efficient but faster

Hybrid metaheuristics (HM) variants

- HM(UB), HMb(UB), MS(UB)
    - decoding step limited to computation of upper bound
    - no computation of lower bound and labels
    - less efficient but faster

Table 5
Solution values.

Instance	LB	Method									
		HM	HMb	MS	HM(LL)	HMb(LL)	MS(LL)	HM(UB)	HMb(UB)	MS(UB)	B&C
CMT1 (50,3,2)	145.86	168	168	188	**166**	168	196	174	174	204	188
CMT2 (75,5,5)	101.54	**130.23**	133.60	148	132	133.41	152	140	140	152	3630.86
CMT3 (100,4,4)	160.86	**184**	186	208	187.04	186.17	204	195.42	197.24	216	4537.11
CMT4 (150,6,6)	115.62	**160.38**	150	184	162	162	180	166	164	192	
CMT5 (199,9,8)	72.65	**138**	139.29	152	140	**138**	154	142.04	140	152	
E-n51-k5 (50,3,2)	145.86	**168**	**168**	182	**168**	**168**	180	168.86	174	196	188
E-n76-k8 (75,4,4)	126.95	**154**	**154**	168	156	156	182	161.86	174	196	2975.51
E-n101-k8 (100,4,4)	160.86	186	**184**	208	188	190.17	224	196	196	216	4537.11
M-n151-k12 (150,6,6)	116.23	**154**	158.96	186	164	162	182	168	169.88	182	
M-n200-k16 (199,8,8)	80.69	**144**	148	162	148	146	156	152	152	168	
P-n51-k10 (50,5,5)	81.34	**111.07**	114	118	112.69	114	122	118	118	133.25	230
P-n55-k7 (54,4,3)	101.46	128	128	138	128	**126**	142	130	132	148	308
P-n60-k10 (59,5,5)	84.30	**114**	116	124	114.86	116	124	122	120	124	246
P-n65-k10 (64,5,5)	94.62	**126**	**126**	138	128	**126**	142	134	131.36	154	580
P-n70-k10 (69,5,5)	99.23	129.29	**128**	138	136	132	146	138	136.56	158	3166.25
P-n76-k5 (75,3,2)	181.34	202	**200**	214	202	202	243.44	210	210	258	280
P-n101-k4 (100,2,2)	321.74	342.69	**342**	396	346	348	388	353.26	354	422	4725.47
X-n110-k13 (109,7,6)	1189.78	**1864**	1898	2080	1898	1898	2044	1926	1960	1970	
X-n115-k10 (114,5,5)	1676.05	**2258**	2300	2658	2262	2274	2504	2316	2332	2862	
X-n139-k10 (138,5,5)	1582.46	2928.64	2740	3144	2534	**2492**	2696	2594	2550	3022	

Conclusion

- hybrid metaheuristics
    - first paper with multiple trucks (PDSMTSP)
    - no comparison with literature possible
- branch-and-cut
    - no competitive results

Future work

- branch-and-price
- better compromises between decoding quality and number of labels in decoding step
- Constraint programming framework
- more realistics models (time windows, drone recharching)
- investigate different ratios for drone and truck fleet sizes