The 5th EUROYoung Workshop - Naples 2025

Exact and Machine Learning-Guided Matheuristic Approaches for the Flying Sidekick TSP

Maurizio Boccia

maurizio.boccia@unina.it

Department of Electrical Engineering and Information Technology,
Optimization and Problem Solving Laboratory
University of Naples Federico II

DIETI UNINA

OPS Lab
Optimization and Problem Solving Laboratory
Laboratorio di Ottimizzazione e Problem Solving

ACHILLES
ecosustAinable effiCient tecH-drIven Last miLE logiStics
PRIN Project

MOST
CENTRO NAZIONALE PER LA MOBILITÀ SOSTENIBILE

Drones in Last-Mile Logistics: Research and Projects at OPSLab

Over the past years, the OPSLab at DIETI-UNINA has focused on optimization problems related to the use of drones in last-mile logistics. The lab is currently involved in three national research projects addressing these topics.

Selected Publications

- ✓ *M. Boccia, A. Masone, A. Sforza & C. Sterle. A column and row generation approach for the FSTSP. TR-C, 2021*
- ✓ *A. Masone, S. Poikonen & B. Golden. The multivisit drone routing problem with edge launches: An iterative approach with discrete and continuous improvements. Networks, 2022.*
- ✓ *M. Boccia, A. Mancuso, A. Masone, & C. Sterle. A new MILP formulation for the FSTSP. Networks, 2023.*
- ✓ *M. Boccia, A. Mancuso, A. Masone, T. Murino & C. Sterle. New features for customer classification in the FSTSP. ESWA, 2024*
- ✓ *M. Boccia, A. Masone, A. Sforza & C. Sterle. Exact and heuristic approaches for the Truck–Drone Team Logistics Problem. TR-C, 2024*
- ✓ *D. Amitrano, M. Boccia, A Masone & C. Sterle. A New Formulation for the Traveling Salesman Problem With Drone and Lockers. Networks, 2025.*
- ⟳ ⟳ *C. Archetti, M. Boccia, A Masone & C. Sterle. A new MILP formulation and a Matheuristic approach for the TSP with Release Dates and Drone Resupply. Submitted to an international journal.*
- ⟳ ⟳ *M. Boccia, M. Brambilla, R. Mansini, A Masone & C. Sterle. A Machine Learning-Guided Matheuristic Approaches for the Flying Sidekick TSP. Ongoing work*

Current Research Projects

- ➢ *ACHILLES – PRIN2022 project on eco-sustainable efficient tech-driven last-mile logistics, funded by EU–Next Generation EU and MUR, led by UniNa with IASI-CNR and UniCt.*

ACHILLES

ecosustAinable efficient tecH-drIven Last miLE logiStics

- ➢ *COSMO – PRIN2022 project on Co-Opetitive Sustainable Mobility Optimization, funded EU–NextGen and MUR.*

Finanziato dall'Unione europea

NextGenerationEU

- ➢ *MOST - National Center for Sustainable Mobility. It unites academia, industry, and institutions for greener, smarter transport.*

MOST

CENTRO NAZIONALE PER LA MOBILITÀ SOSTENIBILE

Joint work with the other members of ***OPSLab at DIETI-UNINA****: D. Amitrano, A. Mancuso A. Masone and C. Sterle*

Not solely my fault — proudly a group effort.

OPSLab

Optimization and Problem Solving Laboratory

Laboratorio di Ottimizzazione e Problem Solving

Outline of the presentation

1. ***Drones in logistics: motivations, background and literature review***

2. ***The Flying Sidekick Traveling Salesman Problem (FS-TSP)***

   a) ***A Big-M Free MILP Formulation and a Branch-and-Cut Approach***

   b) ***A Branch-and-Price approach based on ng-route relaxation***

   c) ***Branch-and-Cut vs Branch-and-Price approaches***

   d) ***A machine learning-guided matheuristic***

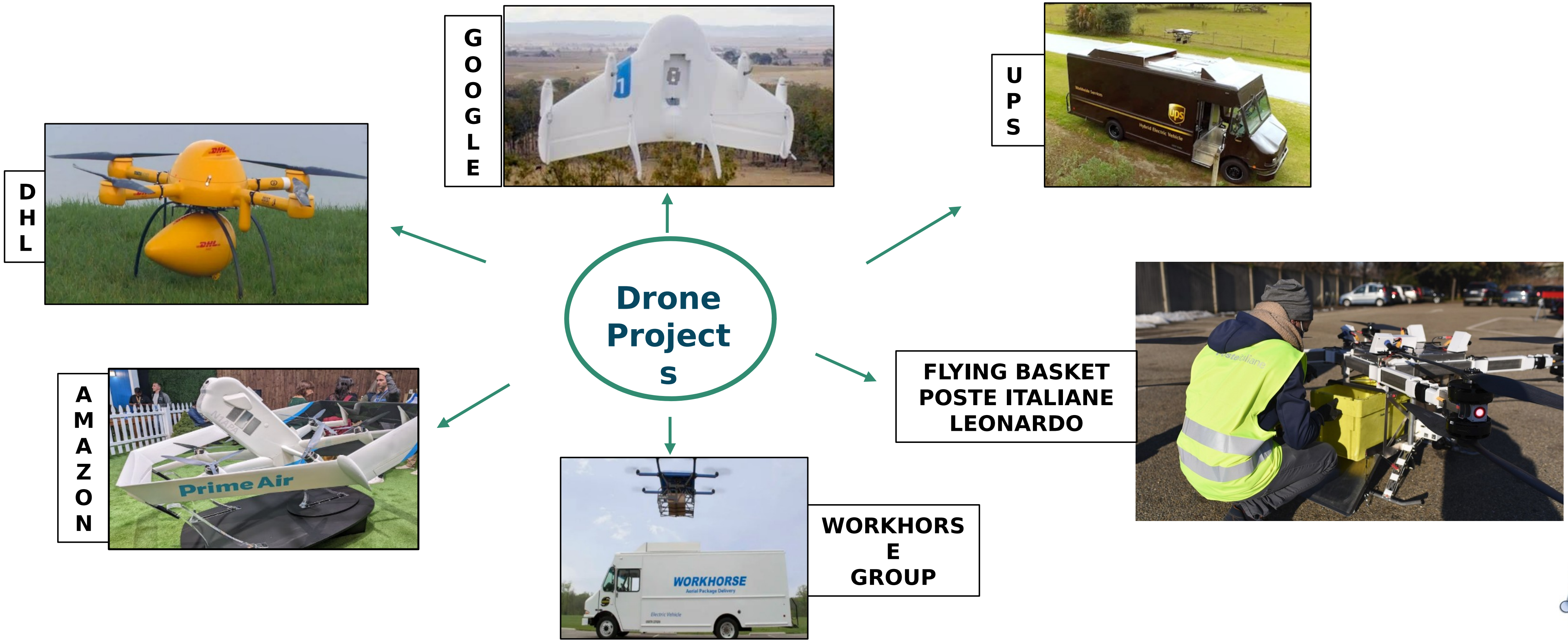
3. ***Conclusions and future research directions***

Drones in logistics: Motivation, Context, and Literature

Literature on Drone in Logistics: multinational use cases

One of the most promising application fields where the use of drones can result useful is the last-mile logistics.

- Several studies showed the benefits, in terms both of emissions and completion time reduction, that can be achieved by using drones for parcel deliveries.



Drone Projects

GOOGLE

UPS

DHL

FLYING BASKET POSTE ITALIANE LEONARDO

AMAZON

WORKHORSE GROUP

Optimization and Problem Solving Laboratory
Laboratorio di Ottimizzazione e Problem Solving

Literature on Drone in Logistics: use cases before 2021

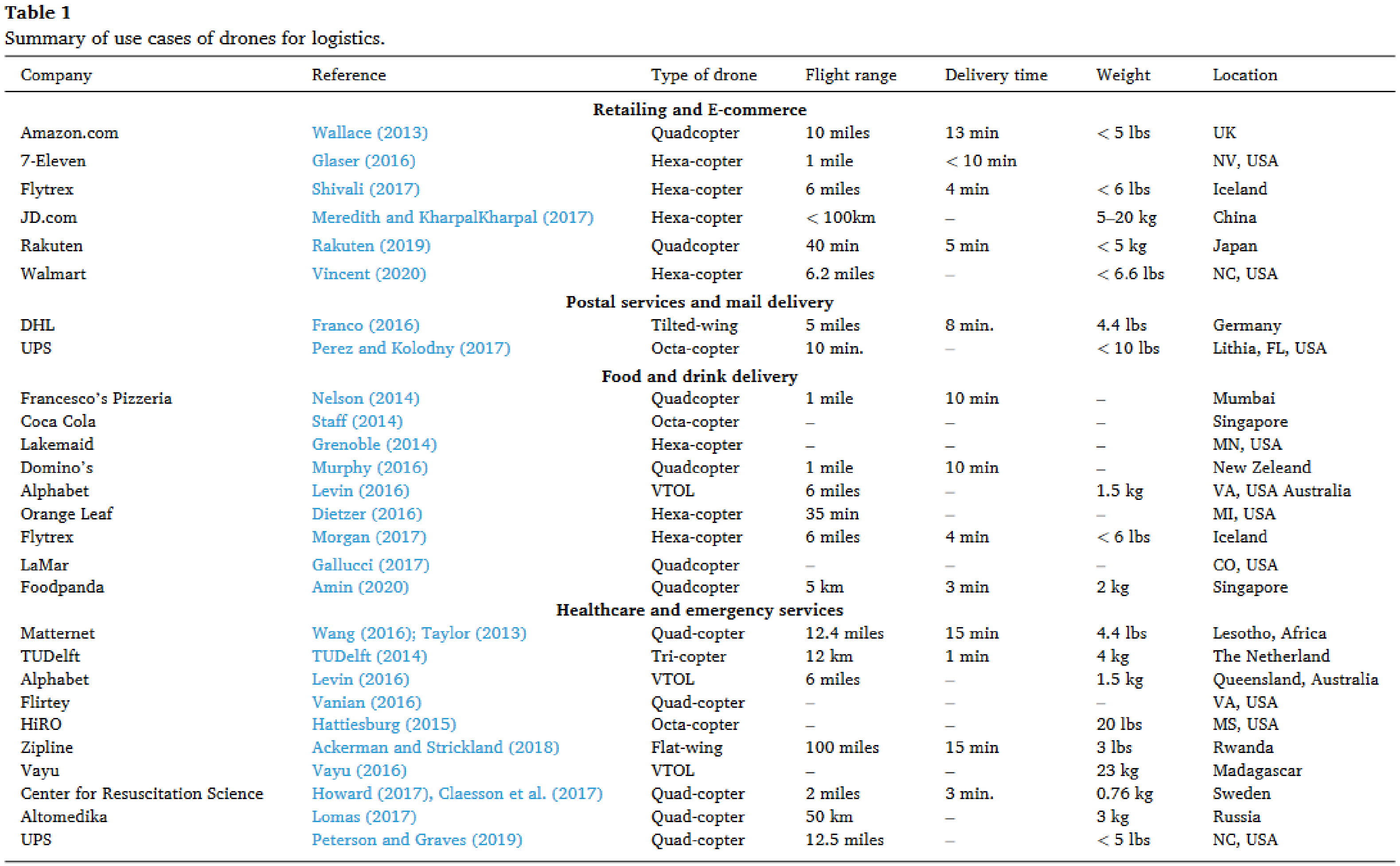


Table 1
Summary of use cases of drones for logistics.

Company	Reference	Type of drone	Flight range	Delivery time	Weight	Location
Retailing and E-commerce						
Amazon.com	Wallace (2013)	Quadcopter	10 miles	13 min	< 5 lbs	UK
7-Eleven	Glaser (2016)	Hexa-copter	1 mile	< 10 min		NV, USA
Flytrex	Shivali (2017)	Hexa-copter	6 miles	4 min	< 6 lbs	Iceland
JD.com	Meredith and KharpalKharpal (2017)	Hexa-copter	< 100km	–	5–20 kg	China
Rakuten	Rakuten (2019)	Quadcopter	40 min	5 min	< 5 kg	Japan
Walmart	Vincent (2020)	Hexa-copter	6.2 miles	–	< 6.6 lbs	NC, USA
Postal services and mail delivery						
DHL	Franco (2016)	Tilted-wing	5 miles	8 min.	4.4 lbs	Germany
UPS	Perez and Kolodny (2017)	Octa-copter	10 min.	–	< 10 lbs	Lithia, FL, USA
Food and drink delivery						
Francesco's Pizzeria	Nelson (2014)	Quadcopter	1 mile	10 min	–	Mumbai
Coca Cola	Staff (2014)	Octa-copter	–	–	–	Singapore
Lakemaid	Grenoble (2014)	Hexa-copter	–	–	–	MN, USA
Domino's	Murphy (2016)	Quadcopter	1 mile	10 min	–	New Zeleand
Alphabet	Levin (2016)	VTOL	6 miles	–	1.5 kg	VA, USA Australia
Orange Leaf	Dietzer (2016)	Hexa-copter	35 min	–	–	MI, USA
Flytrex	Morgan (2017)	Hexa-copter	6 miles	4 min	< 6 lbs	Iceland
LaMar	Gallucci (2017)	Quadcopter	–	–	–	CO, USA
Foodpanda	Amin (2020)	Quadcopter	5 km	3 min	2 kg	Singapore
Healthcare and emergency services						
Matternet	Wang (2016); Taylor (2013)	Quad-copter	12.4 miles	15 min	4.4 lbs	Lesotho, Africa
TUDelft	TUDelft (2014)	Tri-copter	12 km	1 min	4 kg	The Netherland
Alphabet	Levin (2016)	VTOL	6 miles	–	1.5 kg	Queensland, Australia
Flirtey	Vanian (2016)	Quad-copter	–	–	–	VA, USA
HiRO	Hattiesburg (2015)	Octa-copter	–	–	20 lbs	MS, USA
Zipline	Ackerman and Strickland (2018)	Flat-wing	100 miles	15 min	3 lbs	Rwanda
Vayu	Vayu (2016)	VTOL	–	–	23 kg	Madagascar
Center for Resuscitation Science	Howard (2017), Claesson et al. (2017)	Quad-copter	2 miles	3 min.	0.76 kg	Sweden
Altomedika	Lomas (2017)	Quad-copter	50 km	–	3 kg	Russia
UPS	Peterson and Graves (2019)	Quad-copter	12.5 miles	–	< 5 lbs	NC, USA

Moshref-Javadi, M., & Winkenbach, M. (2021). Applications and Research avenues for drone-based models in logistics: A classification and review. Expert Systems with Applications, 177, 114854.

Literature on Drone in Logistics: recent use cases

Company/Project	References	Range	Payload	Sector	Country
Walmart + Wing	*Garg et al. (2023)*	Up to 6 miles	≤ 5 lb	Retail & Grocery	USA
Walmart + Zipline (P2)	*Garg et al. (2023)*	10 miles	6–8 lb	Retail & Grocery	USA
Amazon Prime Air (MK30)	*Garg et al. (2023)*	— (BVLOS)	≤ 5 lb	Retail & General	USA
Manna Drone Delivery	*Sorbelli et al. (2024)*	≈ 3 km	≈ 4 kg	Food & Convenience	Ireland
Swoop Aero – 'Kite'	*Ostermann et al. (2025)*	Up to 120 km	≈ 3 kg	Healthcare	Ethiopia/Malawi/ Australia
Meituan (Keeta)	*Sorbelli et al. (2024)*	Urban routes	< 3 kg	Food & Quick Commerce	China
Flytrex + DoorDash	*Jazairy (2024)*	≈ 5 miles	≤ 3 kg	Food & Grocery	USA
UPS Flight Forward + Matternet M2	*Jazairy (2024)*	Up to 20 km	≈ 2 kg	Healthcare	USA
Wing + NHS/Apian	*Ostermann et al. (2025)*	Urban corridors	—	Healthcare	UK
SF Express	*Ostermann et al. (2025)*	Inter-city trial	—	Express logistics	China

Literature on Drone in Logistics: surveys

- Otto, A., Agatz, N., Campbell, J., Golden, B., & Pesch, E. (*2018*). *Optimization approaches for civil applications of unmanned aerial vehicles (UAVs) or aerial drones: A survey. Networks*, *72*(4), 411-458.
- Khoufi, I., Laouiti, A., & Adjih, C. (*2019*). *A survey of recent extended variants of the traveling salesman and vehicle routing problems for unmanned aerial vehicles. Drones*, *3*(3), 66.
- Macrina, G., Pugliese, L. D. P., Guerriero, F., & Laporte, G. (*2020*). *Drone-aided routing: A literature review. Transportation Research Part C: Emerging Technologies*, *120*, 102762.
- Rojas Viloria, D., Solano-Charris, E. L., Muñoz-Villamizar, A., & Montoya-Torres, J. R. (*2021*). *Unmanned aerial vehicles/drones in vehicle routing problems: a literature review. International Transactions in Operational Research*, *28*(4), 1626-1657.
- Boysen, N., Fedtke, S., & Schwerdfeger, S. (*2021*). *Last-mile delivery concepts: a survey from an operational research perspective. Or Spectrum*, *43*, 1-58.
- Moshref-Javadi, M., & Winkenbach, M. (2021). *Applications and Research avenues for drone-based models in logistics: A classification and review.* Expert Systems with Applications, 177, 114854.
- Sah, B., Gupta, R., & Bani-Hani, D. (*2021*). *Analysis of barriers to implement drone logistics. International Journal of Logistics Research and Applications*, *24*(6), 531-550.
- Liang, Y. J., & Luo, Z. X. (*2022*). *A survey of truck–drone routing problem: Literature review and research prospects. Journal of the Operations Research Society of China*, *10*(2), 343-377.
- Rejeb, A., Rejeb, K., Simske, S. J., & Treiblmaier, H. (2023). *Drones for supply chain management and logistics: a review and research agenda.* International Journal of Logistics Research and Applications, 26(6), 708-731.
- Garg, V., Niranjan, S., Prybutok, V., Pohlen, T., & Gligor, D. (2023). *Drones in last-mile delivery: A systematic review on Efficiency, Accessibility, and Sustainability.* Transportation Research Part D: Transport and Environment, 123, 103831.
- Jahani, H., Khosravi, Y., Kargar, B., Ong, K. L., & Arisian, S. (2025*). Exploring the role of drones and UAVs in logistics and supply chain management: a novel text-based literature review.* International Journal of Production Research, 63(5), 1873-1897.

OPSLab
Optimization and Problem Solving Laboratory
Laboratorio di Ottimizzazione e Problem Solving

Last mile delivery with truck and drone systems

The best approach for last-mile logistics involves the combined use of traditional vehicles and drones. (Agats et al., Transp. Sci.2018; Chung et al. Comp. Oper. Res. 2020; Liang and Luo, J. Oper. Res. Soc. China 2022; etc.)

The main delivery systems using a truck and a drone for parcel transportation can be classified based on two key parameters:

Synchronization between the truck and the drone:

- the two vehicles operate *in parallel with no interaction* between them
- the two vehicles operate *in tandem* coordinating their movements

Interaction between the drone and the customers:

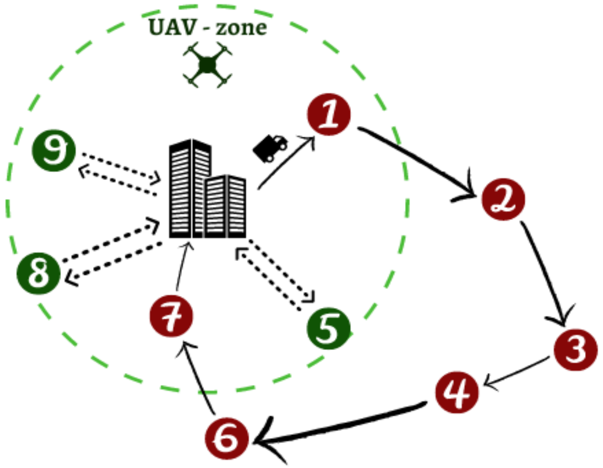
- the drone delivers parcels *directly to customer locations*
- the drone *resupplies the truck with parcels* becoming available while deliveries are in progress

		Truck-Drone Synchronization	
		Yes	No
Drone-Customer Interaction	Yes		
	No		

Based on these two parameters, we can identify three different truck-and-drone systems for last-mile delivery

Last mile delivery with truck and drone systems

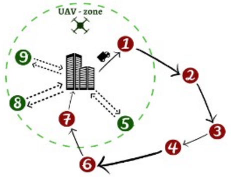
Parallel truck-drone system



- the two vehicles operate *in parallel with no interaction* between them
- the drone delivers parcels *directly to customer locations*

Parallel Drone Scheduling TSP

C. C. Murray and A. G. Chu. *The Flying Sidekick Traveling Salesman Problem: Optimization of Drone-assisted Parcel Delivery*. *Transportation Research Part C, 2015.*

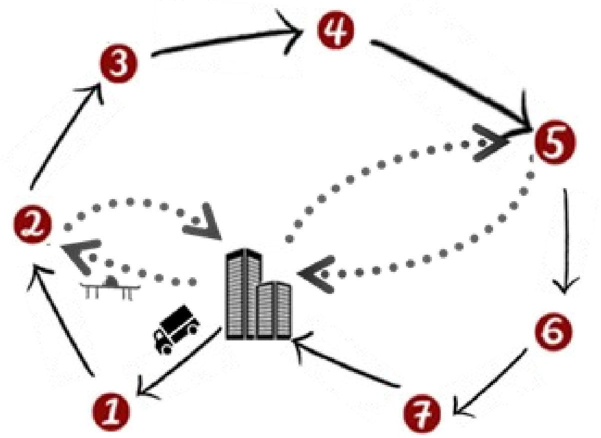
		Truck-Drone Synchronization	
		Yes	No
Drone-Customer Interaction	Yes		***PDS- TSP*** 
	No		

Truck movement ⟶

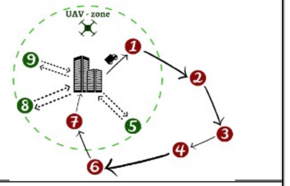
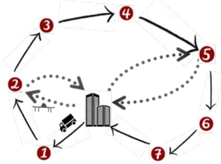
Drone movement - - - →

Last mile delivery with truck and drone systems

Drone resupply system



- the two vehicles operate *in tandem* coordinating their movements
- there is *no interaction* between the drone and the customers
- not all customer parcels are available at the depot at the beginning of the planning horizon
- the *drone resupplies the truck* with parcels released during ongoing deliveries

		Truck-Drone Synchronization	
		Yes	No
Drone-Customer Interaction	Yes		***PDS- TSP*** 
	No	***TSP-RD-DR*** 	

TSP with Release Dates and Drone Resupply

J. C. Pina-Pardo, D. F. Silva, & A. E. Smith. *The traveling salesman problem with release dates and drone resupply. Computers & Operations Research*, 2021.

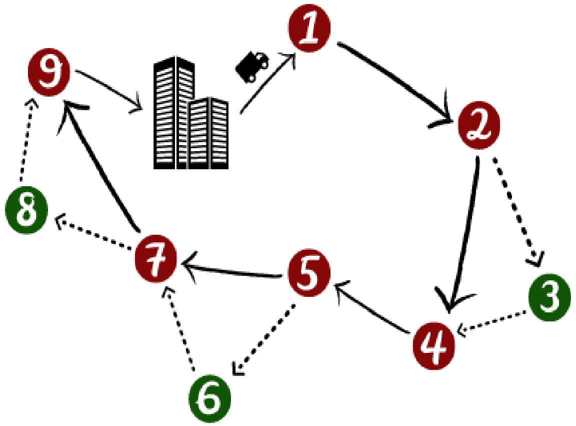
Truck movement ⟶

Drone movement - - - →

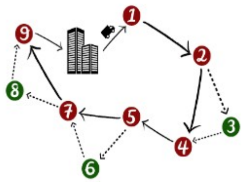
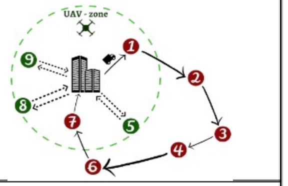
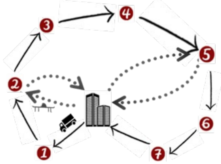
Optimization and Problem Solving Laboratory
Laboratorio di Ottimizzazione e Problem Solving

Last mile delivery with truck and drone systems

Synchronized truck-drone system



- the two vehicles operate *in tandem* coordinating their movements
- the drone delivers parcels *directly to customer locations*

		Truck-Drone Synchronization	
		Yes	No
Drone-Customer Interaction	Yes	***FS-TSP*** 	***PDS- TSP*** 
	No	***TSP-RD-DR*** 	

Flying Sidekick TSP

C. C. Murray and A. G. Chu. *The Flying Sidekick Traveling Salesman Problem: Optimization of Drone-assisted Parcel Delivery. Transportation Research Part C, 2015.*

Truck movement ——→

Drone movement - - - →

Synchronized truck and drone system

How truck and drone work together to optimize deliveries

The Flying Sidekick Traveling Salesman Problem (FS-TSP)

FS-TSP: problem statement

The Flying Sidekick TSP is a variant of the traveling salesman problem involving drone-assisted parcel delivery

The FS-TSP is a delivery problem in which a truck and a drone (unmanned aerial vehicle UAV) are used as a synchronized working unit to make the delivery process more efficient and, possibly, less expensive.

Drone tasks

- ✓ *the drone is launched from the truck,*
- ✓ *it proceeds to deliver goods to a customer*
- ✓ *it joins back the truck before its endurance is exhausted*

Truck tasks

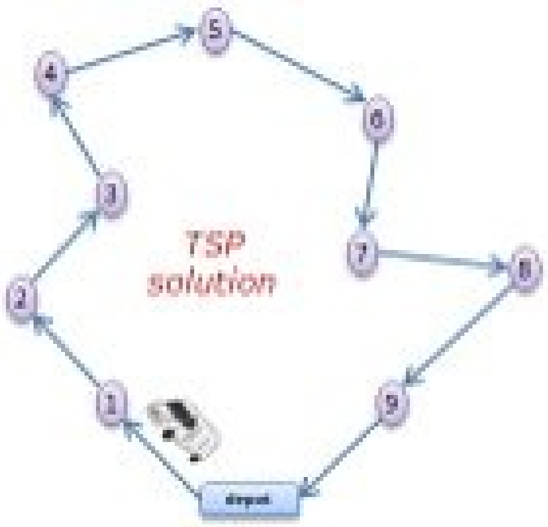
- ✓ *while the drone is serving a customer, the truck travels to deliver parcels to other customers*
- ✓ *it has to recover the drone before finishing the endurance of the drone*
- ✓ *it can serve other customers when the drone is "on board"*

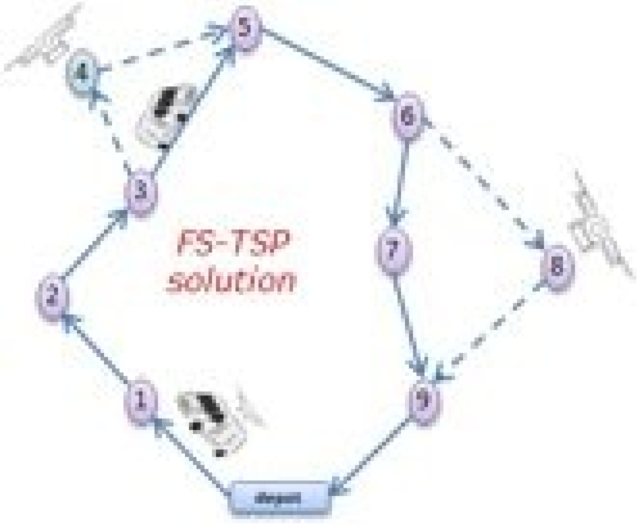
FS-TSP: problem statement

The Flying Sidekick TSP is a variant of the traveling salesman problem involving drone-assisted parcel delivery

The FS-TSP is a delivery problem in which a truck and a drone (unmanned aerial vehicle UAV) are used as a synchronized working unit to make the delivery process more efficient and, possibly, less expensive.

Example:





Serving two customers with the drone instead of the truck can reduce the overall delivery time required to serve all the customers.

FS-TSP: problem statement

The Flying Sidekick TSP is a variant of the traveling salesman problem involving drone-assisted parcel delivery

The FS-TSP is a delivery problem in which a truck and a drone (unmanned aerial vehicle UAV) are used as a synchronized working unit to make the delivery process more efficient and, possibly, less expensive.

FS-TSP decision levels:

- ***Routing decision*** → *route performed by the truck*
- ***Assignment decision*** → *clients to served by the drone or by the truck*
  (the drone can only deliver parcels to eligible customers – payload capacity)
- ***Operational Decisions*** → *launch and pick-up node for each drone flight*
  (considering the endurance limit of the drone)

Objective: ***minimize the overall delivery time***

FS-TSP: literature review

Flying Sidekick TSP: exact approaches

- C. C. Murray & A. G. Chu. *The Flying Sidekick Traveling Salesman Problem: Optimization of Drone-assisted Parcel Delivery*. *Transportation Research Part C, 2015*.
  - *MILP formulation and two heuristic approaches*
  - *None of the 10 customers benchmark instances solved to optimality.*
- M. Dell'Amico, R. Montemanni & S. Novellani. *Exact models for the flying sidekick traveling salesman problem*. *International Transactions in Operational Research, 2022*.
  - *Branch and cut approach*
  - *Solve to optimality most of the 20 customers Murray instances*
- M. Boccia, A. Masone, A. Sforza & C. Sterle. *A column and row generation approach for the flying sidekick traveling salesman problem*. *Transportation research Part C: Emerging Technologies, 2021*.
  - *Column and row generation approach*
  - *Solve to optimality most of the 20 customers Murray instances*
- R. Roberti & M. Ruthmair. *Exact methods for the traveling salesman problem with drone. Transportation Science, 2021*.
  - *Branch and price and dynamic programming*
  - *Solve to optimality most of the 40 customers Poikonen instances*
- M. Boccia, A. Mancuso, A. Masone, & C. Sterle. *A new MILP formulation for the flying sidekick traveling salesman problem*. *Networks, 2023*.
  - *Branch and Cut approach*
  - *Solve to optimality the 20 customers Murray instances*
  - *Solve to optimality most of instances up to 40 customers Poikonen instances*
- M. Blufstein, G. Lera-Romero & F. J. Soulignac. *Decremental State-Space Relaxations for the Basic Traveling Salesman Problem with a Drone*. *INFORMS Journal on Computing, 2024*.
  - *Branch and Price and dynamic programming approach*
  - *Solve to optimality instances up to 60 customers Poikonen instances*

FS-TSP: drawbacks in literature models

To account for ***synchronization constraints*** between drone and truck, most of the models in the literature either contain *Big-M constraints* or use an *exponential number of variables*.

Murray and Chu (2015)

Decision variables:

- *if the arc (i,j) is included to the truck route*
- sortie *if the drone performs the sortie*
- *time* *time at which the truck (drone) arrives at node j*
- *precedence* *if customer i is visited before costomer j*
- *p* *specifies the position of node i in the truck route*

FS-TSP: drawbacks in literature models

To account for ***synchronization constraints*** between drone and truck, most of the models in the literature either contain *Big-M constraints* or use an *exponential number of variables*.

Murray and Chu (2015)

$$\text{Min} \quad t_{c+1}$$

$$\text{s.t.} \quad \sum_{\substack{i\in N_0\\ i\neq j}} x_{ij} + \sum_{\substack{i\in N_0\\ i\neq j}} \sum_{\substack{k\in N_+\\ \langle i,j,k\rangle\in P}} y_{ijk} = 1 \quad \forall j \in C$$

$$\sum_{j\in N_+} x_{0j} = 1$$

$$\sum_{i\in N_0} x_{i,c+1} = 1$$

$$u_i - u_j + 1 \leqslant (c+2)(1-x_{ij})$$
$$\forall i \in C, j \in \{N_+ : j \neq i\}$$

$$\sum_{\substack{i\in N_0\\ i\neq j}} x_{ij} = \sum_{\substack{k\in N_+\\ k\neq j}} x_{jk} \quad \forall j \in C$$

$$\sum_{\substack{j\in C\\ j\neq i}} \sum_{\substack{k\in N_+\\ \langle i,j,k\rangle\in P}} y_{ijk} \leqslant 1 \quad \forall i \in N_0$$

$$\sum_{\substack{i\in N_0\\ i\neq k}} \sum_{\substack{j\in C\\ \langle i,j,k\rangle\in P}} y_{ijk} \leqslant 1 \quad \forall k \in N_+$$

$$2y_{ijk} \leqslant \sum_{\substack{h\in N_0\\ h\neq i}} x_{hi} + \sum_{\substack{l\in C\\ l\neq k}} x_{lk}$$
$$\forall i \in C, j \in \{C : j \neq i\}, k \in \{N_+ : \langle i,j,k\rangle \in P\}$$

$$y_{0jk} \leqslant \sum_{\substack{h\in N_0\\ h\neq k}} x_{hk} \quad \forall j \in C, k \in \{N_+ : \langle 0,j,k\rangle \in P\}$$

$$u_k - u_i \geqslant 1 - (c+2)\left(1 - \sum_{\substack{j\in C\\ \langle i,j,k\rangle\in P}} y_{ijk}\right)$$
$$\forall i \in C, k \in \{N_+ : k \neq i\}$$

$$t'_i \geqslant t_i - M\left(1 - \sum_{\substack{j\in C\\ j\neq i}} \sum_{\substack{k\in N_+\\ \langle i,j,k\rangle\in P}} y_{ijk}\right) \quad \forall i \in C$$

$$t'_i \leqslant t_i + M\left(1 - \sum_{\substack{j\in C\\ j\neq i}} \sum_{\substack{k\in N_+\\ \langle i,j,k\rangle\in P}} y_{ijk}\right) \quad \forall i \in C$$

$$t'_k \geqslant t_k - M\left(1 - \sum_{\substack{i\in N_0\\ i\neq k}} \sum_{\substack{j\in C\\ \langle i,j,k\rangle\in P}} y_{ijk}\right) \quad \forall k \in N_+$$

$$t'_k \leqslant t_k + M\left(1 - \sum_{\substack{i\in N_0\\ i\neq k}} \sum_{\substack{j\in C\\ \langle i,j,k\rangle\in P}} y_{ijk}\right) \quad \forall k \in N_+$$

$$t_k \geqslant t_h + \tau_{hk} + s_L\left(\sum_{\substack{l\in C\\ l\neq k}} \sum_{\substack{m\in N_+\\ \langle k,l,m\rangle\in P}} y_{klm}\right) + s_R\left(\sum_{\substack{i\in N_0\\ i\neq k}} \sum_{\substack{j\in C\\ \langle i,j,k\rangle\in P}} y_{ijk}\right) - M(1 - x_{hk})$$
$$\forall h \in N_0, k \in \{N_+ : k \neq h\}$$

$$t'_j \geqslant t'_i + \tau'_{ij} - M\left(1 - \sum_{\substack{k\in N_+\\ \langle i,j,k\rangle\in P}} y_{ijk}\right) \quad \forall j \in C', i \in \{N_0 : i \neq j\}$$

$$t'_k \geqslant t'_j + \tau'_{jk} + s_R - M\left(1 - \sum_{\substack{i\in N_0\\ \langle i,j,k\rangle\in P}} y_{ijk}\right) \quad \forall j \in C', k \in \{N_+ : k \neq j\}$$

$$t'_k - (t'_j - \tau'_{ij}) \leqslant e + M(1 - y_{ijk}) \quad \forall k \in N_+, j \in \{C : j \neq k\}, i \in \{N_0 : \langle i,j,k\rangle \in P$$

$$u_i - u_j \geqslant 1 - (c+2)p_{ij} \quad \forall i \in C, j \in \{C : j \neq i\}$$

$$u_i - u_j \leqslant -1 + (c+2)(1 - p_{ij}) \quad \forall i \in C, j \in \{C : j \neq i\}$$

$$p_{ij} + p_{ji} = 1 \quad \forall i \in C, j \in \{C : j \neq i\}$$

$$t'_l \geqslant t'_k - M\left(3 - \sum_{\substack{j\in C\\ \langle i,j,k\rangle\in P\\ j\neq l}} y_{ijk} - \sum_{\substack{m\in C\\ m\neq i\\ m\neq k\\ m\neq l}} \sum_{\substack{n\in N_+\\ \langle l,m,n\rangle\in P\\ n\neq i\\ n\neq k}} y_{lmn} - p_{il}\right)$$
$$\forall i \in N_0, k \in \{N_+ : k \neq i\}, l \in \{C : l \neq i, l \neq k\}$$

$$t_0 = 0$$

$$t'_0 = 0$$

$$p_{0j} = 1 \quad \forall j \in C$$

$$x_{ij} \in \{0,1\} \quad \forall i \in N_0, j \in \{N_+ : j \neq i\}$$

$$y_{ijk} \in \{0,1\} \quad \forall i \in N_0, j \in \{C : j \neq i\}, k \in \{N_+ : \langle i,j,k\rangle \in P\}$$

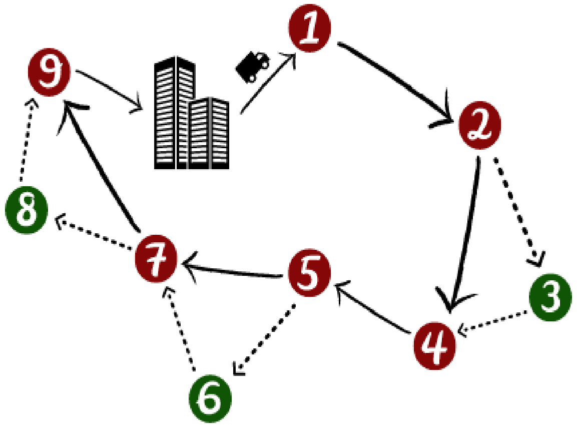
$$1 \leqslant u_i \leqslant c+2 \quad \forall i \in N_+$$

$$t_i \geqslant 0 \quad \forall i \in N$$

$$t'_i \geqslant 0 \quad \forall i \in N$$

$$p_{ij} \in \{0,1\} \quad \forall i \in N_0, j \in \{C : j \neq i\}.$$

A MILP formulation without big-M constraints and with a polynomial number of variables



FS-TSP: MILP formulation

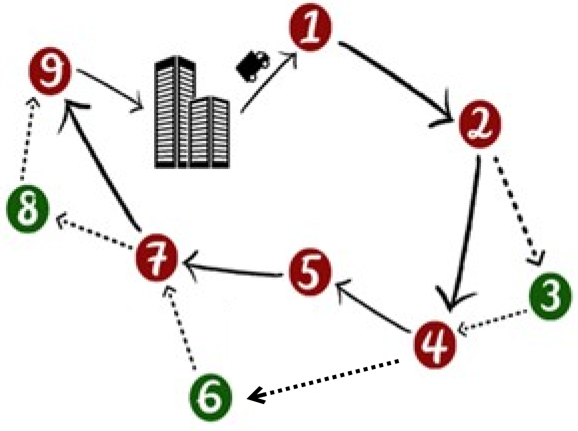
Information fully defining a feasible solution

Truck information

- ✓ *The truck route*
- ✓ *The total travel time along the arcs of the route*

Drone sorties (for each sortie):

- ✓ *The drone path and its total travel time*
- ✓ *The truck path and its total travel time*
- ✓ *The truck waiting time, computed as the difference between the drone travel time and the truck travel ti*



The objective value of the solution is given by the sum of:

- ✓ *The truck travel time*
- ✓ *The service time for launch and recovery of each sortie*
- ✓ *The truck waiting time for each sortie*

A solution can be fully defined without requiring the exact arrival times of the truck and the drone at the nodes.

FS-TSP: MILP formulation

- The depot is splitted into two nodes, an origin node () with only outgoing arcs and a destination node () with only incoming arcs.
- is the complete directed graph where and is the set of arcs.
- , is the truck travelling time and is the drone travelling time.
- and are the drone launch and recovery time, respectively.
- is the drone time limit.

Binary decision variables:

- if the arc (i,j) belongs to the truck path
- if the arc (i,j) is crossed by the truck while the drone is serving
- =1 if client h is served by the drone
- if node i is the origin node of the sortie serving client
- if node i is the destination node of the sortie serving client

Continuous decision variables:

- is the waiting time of the truck at the destination node of the sortie serving client

FS-TSP: MILP formulation

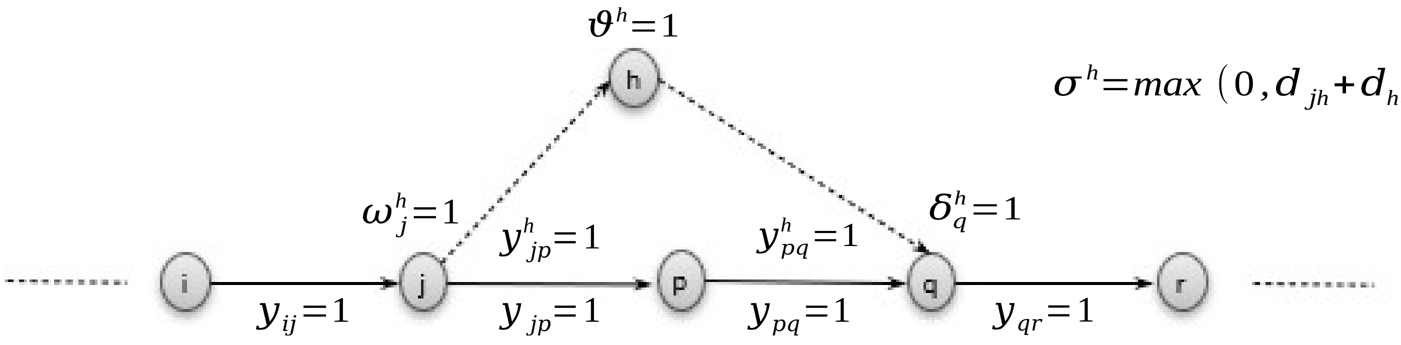
Binary decision variables:

- if the arc (i,j) belongs to the truck path
- if the arc (i,j) is crossed by the truck while the drone is serving
- = 1 if client h is served by the drone
- if node i is the origin node of the sortie serving client
- if node i is the destination node of the sortie serving client

Continuous decision variables:

- is the waiting time of the truck at the destination node of the sortie serving client

Example of feasible solution



$$\sigma^h = max\left(0, d_{jh} + d_{hq} - t_{jp} - t_{pq}\right)$$

FS-TSP: MILP formulation

Binary decision variables:

- *if the arc (i,j) belongs to the truck path*
- *if the arc (i,j) is crossed by the truck while the drone is serving*
- = 1 *if client h is served by the drone*
- *if node i is the origin node of the sortie serving client*
- *if node i is the destination node of the sortie serving client*

Continuous decision variables:

- *is the waiting time of the truck at the destination node of the sortie serving client*

Objective function

$$Min \sum_{(i,j)\in A} t_{ij} y_{ij} + \sum_{h\in C} (SL+SR)\vartheta^{h} - \sum_{h\in C} S\,L\,\omega_{s}^{h} + \sum_{h\in C} \sigma^{h}$$

Truck Route Length

Launch and Recovery Time

Waiting Time

FS-TSP: MILP formulation

Truck routing constraints:

$$\sum_{j:(s,j)\in A} y_{sj} = \sum_{i:(i,t)\in A} y_{it} = 1$$

One outgoing arc from the origin and one ingoing arc to the destination node

$$\sum_{j:(i,j)\in A} y_{ij} = \sum_{j:(j,i)\in A} y_{ji} \leq 1 \, i \in V$$

If i is visited by the truck, it must have exactly one incominc and one outgoing arc

$$\sum_{i,j\in S \vee (i,j)\in A} y_{ij} \leq \sum_{h\in S ¿ q\}} \left(1-\vartheta^h\right) S \subseteq V, q \in S$$

Subtour elimination constraints

Truck sortie constraints:

$$\sum_{j:(i,j)\in A} y_{ij}^h - \sum_{j:(j,i)\in A} y_{ji}^h = \omega_i^h - \delta_i^h \, i \in V \, h \in C$$

For each sortie, there is a path from the launch node to the recovery node travelled by the truck without the drone on board

Single assignment constraints:

$$\sum_{j:(h,j)\in A} y_{hj} + \vartheta^h = 1 \, h \in C$$

Each client must be served either by the truck or by the drone

$$y_{ij} + \omega_i^j + \delta_i^j \leq 1 \, (i,j) \in A$$

If (i,j) is crossed by the truck then the drone can’t fly from i to j or from j to i

FS-TSP: MILP formulation

Drone endurance constraints:

The duration of the truck path of a sortie can’t exceed the drone endurance

$$\sum_{(i,j)\in A} t_{ij} y_{ij}^{h} \leq \left(Dtl - SR\right)\vartheta^{h} h \in C$$

The duration of a drone sortie can’t exceed the drone endurance

$$\sum_{i\in V \text{¿} t\}} d_{ih}\omega_{i}^{h} + \sum_{j\in V \text{¿} s\}} d_{hj}\delta_{j}^{h} \leq \left(Dtl - SR\right)\vartheta^{h} h \in C$$

Waiting time constraints:

The truck waiting time of a sortie is given by the difference between the duration of the drone path and the duration of the truck path, if it is greater than 0

$$\sum_{i\in V \text{¿} t\}} d_{ih}\omega_{i}^{h} + \sum_{j\in V \text{¿} s\}} d_{hj}\delta_{j}^{h} - \sum_{(i,j)\in A} t_{ij} y_{ij}^{h} \leq \sigma^{h} h \in C$$

Consistency constraints:

The truck path during a drone sortie must be a subpath or the origin destination path

$$\sum_{h\in C} y_{ij}^{h} \leq y_{ij} \left(i , j\right) \in A$$

If h is served by the drone, the corresponding sortie must have a launch and a recovery node

$$\sum_{i\in V \text{¿} t,h\}} \omega_{i}^{h} = \sum_{j\in V \text{¿} s,h\} \delta_{j}^{h} = \vartheta^{h} h\in C \text{¿}} \text{¿}$$

If i is the launch node of a sortie, it can’t be served by the drone

$$\sum_{h\in C \text{¿} i\}} \omega_{i}^{h} + \vartheta^{i} \leq 1\, i \in C$$

If j is the recovery node of a sortie, it can’t be served by the drone

$$\sum_{h\in C \text{¿} j\}} \delta_{j}^{h} + \vartheta^{j} \leq 1\, j \in C$$

FS-TSP: MILP formulation

Objective function:

$$Min \sum_{(i,j) \in A} t_{ij} y_{ij} + \sum_{h \in C} (SL+SR) \vartheta^h - \sum_{h \in C} SL\, \omega_s^h + \sum_{h \in C} \sigma^h$$

Truck routing constraints:

$$\sum_{j:(s,j) \in A} y_{sj} = \sum_{i:(i,t) \in A} y_{it} = 1$$

$$\sum_{j:(i,j) \in A} y_{ij} = \sum_{j:(j,i) \in A} y_{ji} \leq 1 i \in V$$

$$\sum_{i,j \in S \vee (i,j) \in A} y_{ij} \leq \sum_{h \in S ¿ q\}} (1-\vartheta^h) S \subseteq V, q \in S$$

Truck sortie constraints:

$$\sum_{j:(i,j) \in A} y_{ij}^h - \sum_{j:(j,i) \in A} y_{ji}^h = \omega_i^h - \delta_i^h i \in V h \in C$$

Single assignment constraints:

$$y_{ij} + \omega_i^j + \delta_i^j \leq 1 (i,j) \in A$$

$$\sum_{j:(h,j) \in A} y_{hj} + \vartheta^h = 1 h \in C$$

Drone endurance constraints:

$$\sum_{(i,j) \in A} t_{ij} y_{ij}^h \leq (Dtl - SR) \vartheta^h h \in C$$

$$\sum_{i \in V ¿ t\}} d_{ih} \omega_i^h + \sum_{j \in V ¿ s\}} d_{hj} \delta_j^h \leq (Dtl - SR) \vartheta^h h \in C$$

Waiting time constraints:

$$\sum_{i \in V ¿ t\}} d_{ih} \omega_i^h + \sum_{j \in V ¿ s\}} d_{hj} \delta_j^h - \sum_{(i,j) \in A} t_{ij} y_{ij}^h \leq \sigma^h h \in C$$

Consistency constraints:

$$\sum_{h \in C} y_{ij}^h \leq y_{ij} (i,j) \in A$$

$$\sum_{i \in V ¿ t,h\}} \omega_i^h = \sum_{j \in V i ¿ s,h\}} \delta_j^h = \vartheta^h h \in C ¿ ¿$$

$$\sum_{h \in C ¿ i\}} \omega_i^h + \vartheta \leq 1 i \in C$$

$$\sum_{h \in C ¿ j\}} \delta_j^h + \vartheta^j \leq 1 j \in C$$

Optimization and Problem Solving Laboratory
Laboratorio di Ottimizzazione e Problem Solving

FS-TSP: Branch-and-Cut

Valid inequalities:

In any feasible solution there must be a truck path from the origin node to the generic node and from to the destination node , if is served by the truck ().

Cut inequalities

$$\sum_{(i,j)\in(V_s:V_h)} y_{ij} \geq 1-\vartheta^h h\in C$$

– *cut separating s and h*

$$\sum_{(i,j)\in(V_h:V_t)¿} ¿\, y_{ij} \geq 1-\vartheta^h h\in C$$

– *cut separating h and t*

Cut inequalities strengthen the FS-TSP formulation and they can be used as subtour elimination contraints

- *They are added dynamically to the relaxed formulation (without the subtour elimination constraints).*
- *The separation algorithm consists of the solution of a min-cut problem between and (and), on the graph , where each arc is weighted with the fractional value attained by the variable in the current LP relaxation.*

FS-TSP: Branch-and-Cut, implementation details

Boccia, Mancuso, Masone and Sterle (2023)

The *Branch-and-Cut algorithm* was implemented using the *callbacks* provided by the *Gurobi solver*

- It starts with the solution of the *linear programming relaxation*
- At each node of *the first three levels* of the enumeration tree, the *Cut inequalities* are separated by means of a *max-flow separation procedure* for both integer and fractional solutions.
- At the other nodes of the enumeration tree the *subtour elimination constraints* are separated as *lazy constraints:*
  - whenever an integer solution is found, it is checked whether the constraints are satisfied (i.e., whether the solution contains subtours).
    - *If satisfied, the incumbent value is updated*
    - *If violated, the corresponding subtour elimination constraints are added to the formulation.*

- The experiments were performed on an Intel(R) Core(TM) i7-8700k, 3.70 Ghz, 16.00 GB of RAM.
- The branch-and-cut algorithm has been coded in Python using Gurobi 9.5 Callable Library.

FS-TSP: Branch-and-Cut, computational results

Boccia, Mancuso, Masone and Sterle (2023)

Test bed 1 (Murray and Chu Instances)

- *120 instances with 20 customers and drone endurance Dtl = 20 minutes*
- *120 instances with 20 customers and drone endurance Dtl = 40 minutes*

Best results in Dell'Amico et al. (2022) and in Boccia et al. (2021)

The proposed approach solves to optimality in a time limit of 1 hour :

- ✓ *37 of 38 unsolved instances with Dtl = 20*
- ✓ *102 of 116 unsolved instances with Dtl = 40*

The 2023 results remain the best reported in the literature for this Test bed.

FS-TSP: Branch-and-Cut, computational results

Boccia, Mancuso, Masone and Sterle (2023)

Test bed 2 (Poikonen Instances)

- *300 instances with 9, 19, 29, and 39 custormers, with Dtl = 20.*

Best results (then): Roberti and Ruthmair (2021).

- ✓ *Branch-and-Price approach*
- ✓ *Pricing problem solved with a dynamic programming algorithm*

Results on the Poikonen instances with time limit of 1 hour

	Roberti and Ruthmair	Branch and Cut
Average time	392,8 secs	517 secs.
Average gap	1,37	0,17
# unsolved instances	14	22

FS-TSP: Branch-and-Cut, computational results

Boccia, Mancuso, Masone and Sterle (2023)

Test bed 2 (Poikonen Instances)

- *300 instances with 9, 19, 29, and 39 custormers, with Dtl = 20.*

Best results (then): Roberti and Ruthmair (2022).

Results on the Poikonen instances with time limit of 1 hour

	Roberti and Ruthmair	Branch and Cut
Average time	392,8 secs	517 secs.
Average gap	1,37	0,17
# unsolved instances	14	22

Best current results: Blufstein, Lera-Romero and Soulignac (2024).

- ✓ *Branch-and-Price approach*
- ✓ *Pricing problem solved with a dynamic programming algorithm*
- ✓ *Smart variable fixing strategies*

They are able to solve Poikonen instances with up to 60 nodes.

FS-TSP: Murray and Chu vs. Poikonen Instances

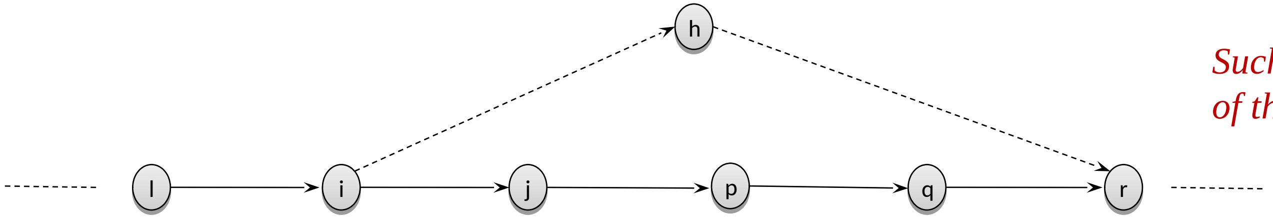
Murray and Chu Instances

- ✓ Customers randomly placed in a 20×20 square
- ✓ *Distribution is not uniform but denser near depot (40%, 60%, and 80% of customers are located within a circle centered at the depot with radius 10 mi.)*
- ✓ Distances: Manhattan (truck), Euclidean (drone).
- ✓ Speeds: Truck = 25 mph, Drone = 25 or 35 mph.
- ✓ SR = SL = 1
- ✓ *Dtl = 20 or 40*

Poikonen Instances

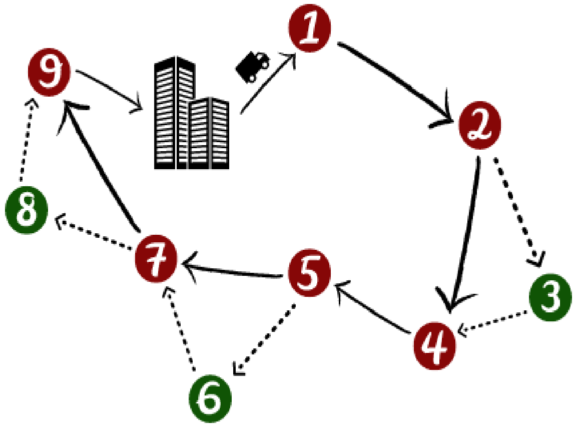
- ✓ Customers and depot randomly placed in a 50×50 grid
- ✓ Truck time = Manhattan distance
- ✓ Drone time = Euclidean distance × α, with α=1,2 or 3
- ✓ No service times (SR = SL = 0)
- ✓ Dtl = 20

Because customers are on average closer to each other, the Murray and Chu instances present a larger number of feasible sorties, often with the truck covering multiple arcs before picking up the drone.



Such sorties are more likely in the feasible solutions of the first test bed than in the second.

A Branch and Price approach for the FS-TSP based on the ng-route relaxation



FS-TSP: set partitioning formulation

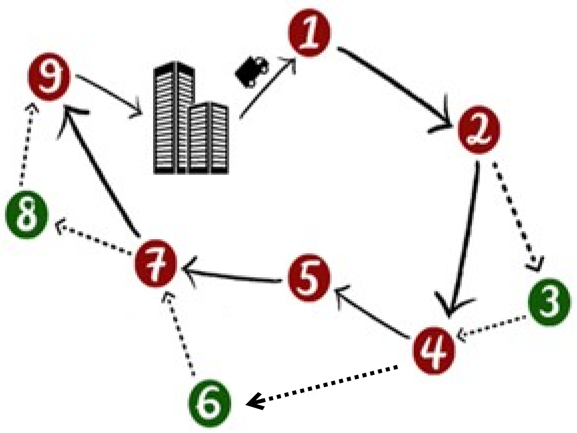
Definitions:

sortie ▭ *is a synchronized pair of a truck path and a two-arc drone path between the same nodes;*

combined path ▭ *is a path followed by the truck with the drone on board;*

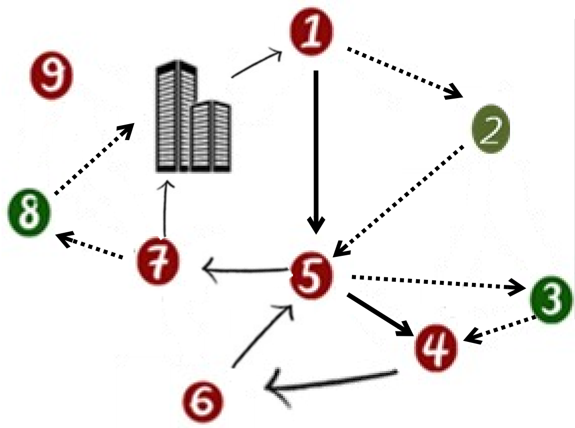
length (of a sortie or combined path) ▭ *is the number of nodes visited, excluding the origin*

*A **route** is an ordered sequence of sorties and combined paths, from the depot back to the depot, with total length ,*
where each component ends at the node where the next one begins



Combined path: d-1-2	*length: 2*
Sortie: 2-4; 1-3-4	*length: 2*
Sortie: 4-5-7; 4-6-7	*length: 3*
Sortie: 7-9; 7-8-9	*length: 2*
Combined path:9-d	*length: 1*

Example of feasible route



Combined path: d-1	*length: 1*
Sortie: 1-5; 1-2-5	*length: 2*
Sortie: 5-4; 5-3-4	*length: 2*
Combined path: 4-6-5-7 *length: 3*	
Sortie:7-d, 7-8-d	*length: 2*

Example of unfeasible route

FS-TSP: set partitioning formulation

- *Let be the set of all route in graph*
- *:*
  - *indicate the number of times customer is visited by route ;*
  - *indicate the duration of route*

Binary decision variables:

- *if route r is selected, otherwise*

Set partitioning Formulation:

$$Min \sum_{r \in R} d_r \xi_r$$

$$\sum_{r \in R} \xi_r = 1$$

Exactly one route must be selected

$$\sum_{r \epsilon R} a_{ir} \xi_r = 1 i \in C$$

Each customer must be visited

$$\xi_r \epsilon \{0,1\} \, r \in R$$

The difficulty of the problem moves from the model to the exponentially large set of variables.

FS-TSP: branch-and-price

At each node of the enumeration tree, a column generation procedure is applied to solve the linear relaxation of the set-partitioning formulation.

Column generation

- *Let be a subset of routes;*
- *Solve the linear relaxation of the subproblem defined by :*
  - *let z* denote the optimal value of the subproblem;*
  - *let be the dual variable associated with the first constraint (;*
  - *let , for , denote the dual variables associated with the partitioning constraints ().*
- *Solve the following **pricing problem**:*

$$\min_{r \in R \setminus R'} d_r - u_0 - \sum_{i \in C} a_{ir} u_r$$

let r be the optimal solution of the pricing problem and d* its value;*

- *If then z* is the optimal value of the complete linear relaxation problem; otherwise update and iterate the procedure.*

FS-TSP: branch-and-price

- *The **pricing problem** on relaxed set can be solved efficiently by dynamic programming.*
- *The set of routes is a relaxation of the set of feasible solutions, but it is far from being tight:*
  - *provides weak lower bounds;*
  - *leads to an exponential growth of the search tree;*
  - *makes medium scale instances unsolvable.*

*A **better approach**: use a tighter relaxation without making the pricing problem too hard to solve.*

*Roberti & Ruthmair (2021) and Blufstein, Lera-Romero & Soulignac (2024) use the **ng-Route relaxation**, first proposed for the truck-only case by Baldacci, Mingozzi & Roberti (2011).*

FS-TSP: branch-and-price, ng-route relaxation

ng-route relaxation: key ideas

➢ *Subtours are allowed* □ *a customer may be skipped or visited more than once.*

➢ *Revisit of node i is allowed only if a «distant» node j is visited in between.*

Definition

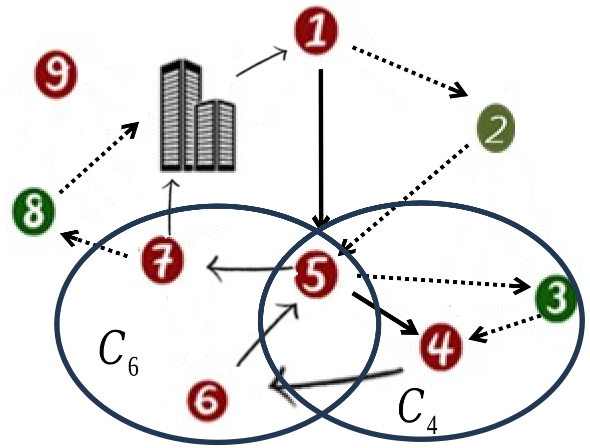
❑ *For each node , the* ***ng-set*** *(is the set of customers closest to ;*

❑ *An* ***ng-route****:*

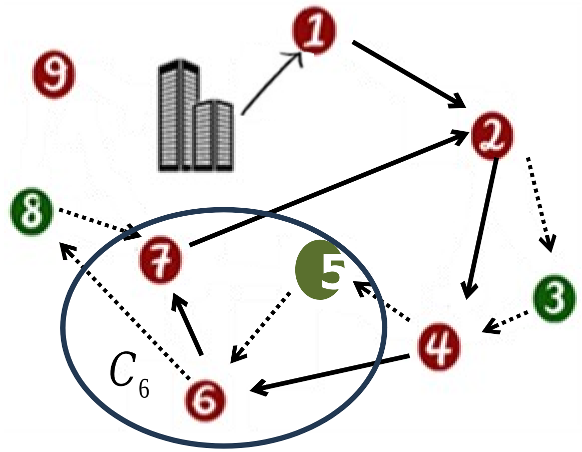
- *starts/ends at the depot*
- *visiting exactly n (not necessary different) nodes*
- *customer can be revisited iff there exists a customer with between the two visits.*

FS-TSP: branch-and-price, ng-route relaxation

Example: non-ng-Route vs. ng-Route



▭ subtour 5->4->6->5 not allowed



$2 \notin C_6$ ▭ subtour 2->4->6->7-2 allowed

- *Subtours in ng-routes, when present, tend to be long and thus not appealing*
- *The lower bound provided by the ng-route relaxation is more effective*

FS-TSP: branch-and-price, ng-route relaxation

- *The **least-reduced-cost ng-route** can be computed by **dynamic programming**.*
  - *Increasing the cardinality of the ng-sets improves the lower bounds provided by the ng-relaxation.*
  - *On the other hand, the larger the cardinality of the ng-sets, the more time-consuming dynamic programming becomes, since it requires more state variables and weaker dominance rules.*

*A proper **trade-off** must be found between the quality of the bounds and the computational effort.*

In M. Blufstein, G. Lera-Romero & F. J. Soulignac. Decremental State-Space Relaxations for the Basic Traveling Salesman Problem with a Drone. INFORMS Journal on Computing, 2024, the authors:

- ❑ *fixed the cardinality of the ng-sets to 5;*
- ❑ *used stronger dominance criteria than Roberti & Ruthmair (2021);*
- ❑ *adopted efficient fixing strategies to speed up the pricing procedure.*

*They achieved **the best results** available in the literature on the Poikonen test bed, **solving to optimality instances with up to 60 nodes**.*

FS-TSP: branch-and-price vs. branch-and-cut

At present, the *state-of-the-art* solution for the FS-TSP relies on a *Branch-and-Price* algorithm proposed by *Blufstein, G. Lera-Romero & F. J. Soulignac (2024).*

The question is:

- Does this really mean that Branch-and-Price is the best way — or even the only way — to solve the FS-TSP?

It is not easy to give a definite straightforward answer.

But to address this question we can take a closer look at how the literature on exact approaches for the FS-TSP has evolved in recent years.

FS-TSP: branch-and-price vs. branch-and-cut

Evolution of exact approaches for the FS-TSP

- ***2015 – CPLEX Branch-and-Cut*** – C. C. *Murray & A. G. Chu. The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery. Transportation Reseach Part C.*

- ***2019 – Branch-and-Cut.*** – *M. Dell'Amico, R. Montemanni & S. Novellani. Drone-assisted deliveries: new formulations for the Flying Sidekick Traveling Salesman Problem. Optimization Letters.*

- **2021 – *Branch-and-Cut-and-Price*** – *M. Boccia, A. Masone, A. Sforza & C. Sterle. A column-and-row generation approach for the flying sidekick traveling salesman problem. Transportation Reseach Part C.*

- ***2022 – Branch-and-Cut*** – *M. Dell'amico, R. Montemanni & S. Novellani. Exact models for the flying sidekick traveling salesman problem. International Trancactions in Operational Research.*

- ***2022* – *Branch-and-Price*** – *R. Roberti & M. Ruthmair. Exact methods for the traveling salesman problem with drone. Transportation Science.*

- ***2023 – Branch-and-Cut*** – *M. Boccia, A. Mancuso, A. Masone & C. Sterle. A new MILP formulation for the flying sidekick traveling salesman problem. Networks.*

- ***2024 – Branch-and-Price*** – *M. Blufstein, G. Lera-Romero & F. Soulignac. Decremental State-Space Relaxations for the Basic Traveling Salesman Problema with a Drone. INFORM Journal an Computing.*

FS-TSP: branch-and-price vs. branch-and-cut

At present, the *state-of-the-art* solution for the FS-TSP relies on a *Branch-and-Price* algorithm proposed by *Blufstein, G. Lera-Romero & F. J. Soulignac (2024).*

The question is:

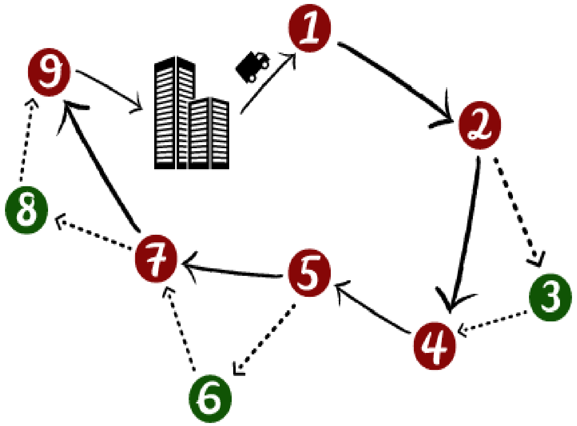
- Does this really mean that Branch-and-Price is the best way — or even the only way — to solve the FS-TSP?

- *If we want to obtain good results from a **Branch-and-Price approach**, the key lies in improving the pricing problem and designing new relaxations to make it more efficient.*
- *On the other hand, if we aim to achieve good performance with a **Branch-and-Cut approach**, we need to focus on polyhedral analysis, the identification of new valid inequalities, as well as effective fixing and branching strategies.*

At present, Blufstein, Lera-Romero, and Soulignac have done an excellent job, working on the pricing problem and achieving the best results so far.

I hope the next time I see you at a conference, it will be because one of our team is presenting a work entitled 'New valid inequalities for the FS-TSP…'— meaning we've finally improved our Branch-and-Cut.

Machine learning-guided matheuristic approach for the FS-TSP



FS-TSP: ML-guided matheuristics

*The solution of a **FS-TSP** instance requires to decide:*

- ✓ *the subsets of customers assigned to the truck and to the drone, respectively;*
- ✓ *the nodes where the drone will be launched and retrieved for each drone sortie*
- ✓ *the order according to which the customers will be visited by the two vehicles*

Key questions:

- ➢ *What information could be useful to reduce the complexity of the problem and its formulation?*

***Drone customers / Truck customers** known in advance* → *problem much simple*
*…but still **NP-hard** (more complex than the TSP)*

- ➢ *But how could we possibly obtain this information?*

When we don’t know the answer…
we rely on Machine Learning!

FS-TSP: ML-guided matheuristics

First matheuristic

Core Idea

- *Goal: reduce the complexity of the original problem.*
- *Approach: use a Machine Learning classifier to assign customers to two categories:*

  Truck only → *can only be served by the truck*

  Drone only → *can only be served by the drone*

Impact on the Formulation

- *Once customers are classified, the FS-TSP formulation can be simplified by fixing a large set of variables.*
- *The reduced model is then solved using the branch-and-cut approach*.

Classification Experiment

- *Input: original customer features.*
- *Output:* label (*truck only / drone only*).
- *Method*: *supervised Machine Learning* experiment.
- *Implementation:* classification models built using the *Scikit-learn package in Python*.

FS-TSP: ML-guided matheuristics

Training set:

Training Set

- Instances generated with *10, 15, and 20 customers*
- *Generation procedure* follows Murray and Chu (2025)
- Parameters varied:
  - ✓ *Customer density* (average number of customers per unit square)
  - ✓ *Drone endurance*
  - ✓ *Drone speed)*

Uniform Dataset Construction

- *Each customer = one observation*
- To *balance the dataset*:
  - ✓ 1,080 instances with 10 customers → 10,800 observations
  - ✓ 720 instances with 15 customers → 10,800 observations
  - ✓ 540 instances with 20 customers → 10,800 observations

Customer features:

Feature Design

- Problem-related features: *parameters from the assumptions*
  - ✓ ***e**.g., drone endurance, drone speed, truck speed, square side lenght …*
- Graph-topology features: *characteristics of the customer node within its local neighborhood*
  - ✓ *e.g., node centrality measures, distance to the nearest customer …*

Modular approach:

- ✓ *Training* is done on *small/medium-sized* instances
- ✓ *Classification* is applied to *larger instances*
- ✓ *Neighborhood features computed only within the subgraph reachable by the drone* (given its endurance)

Overall, we identified 6 problem-related features and 10 graph-topology features, which we then attempted to reduce by applying the feature selection procedures provided by the Scikit-learn.

FS-TSP: ML-guided matheuristics

Customer classification:

We tested eight different classifiers:

- *k-nearest neighbors (KNN),*
- *linear support vector machine (LSV),*
- *kernel support vector machine (RSV),*
- *random forests (RAF),*
- *neural networks (NNE),*
- *adaptive boost algorithm (ADB)*
- *decision tree (DET),*
- *gradient bosting (GBM)*

	test_accuracy	**test_precision**	**test_recall**	**test_f1**
KNN	0.798	0.861	0.884	0.871
LSV	0.848	0.857	0.965	0.908
RSV	0.802	0.830	0.940	0.881
RAF	0.815	0.821	0.978	0.892
NNE	0.852	0.867	0.926	0.896
ADB	0.849	0.870	0.949	0.907
DET	0.851	0.881	0.934	0.907
GRB	0.856	0.887	0.935	0.910

Accuracy → *Among all predictions, the percentage that are correct.*

Precision → *Among the predicted positives, the percentage that are actually positive.*

Recall → *Among the actual positives, the percentage that are correctly predicted.*

F1-Score → *A single measure that balances precision and recall.*

Positive prediction = node classified as truck-only.

Each tested classification model was used with its default parameter settings.

Based on the tests carried out, we chose to use the Gradient Boosting classifier and applied a feature selection procedure, which reduced the initial set from 16 features to 9.

FS-TSP: ML-guided matheuristics

First matheuristic, main steps

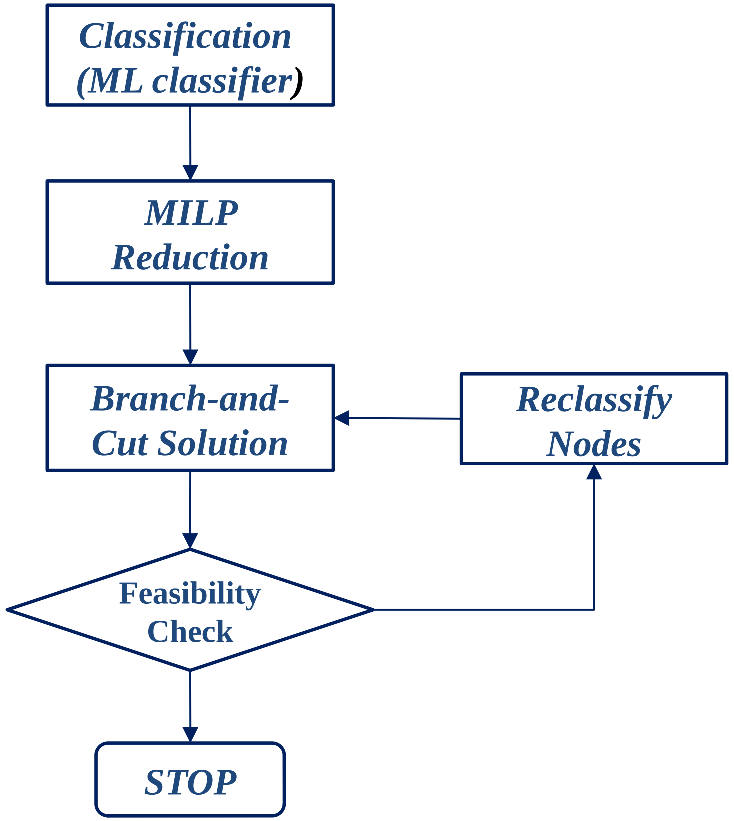
- ***Step 1 – Classification***
  *Nodes are classified into **truck-only** and **drone-only** using the ML classifier.*

- ***Step 2 – MILP Reduction***
  The MILP formulation of the problem is simplified according to this classification.

- ***Step 3 – Solution***
  *The reduced model is solved with the **branch-and-cut** algorithm to obtain a feasible solution.*

- ***Step 4 – Feasibility Check***
  If the reduced model turns out to be infeasible, some nodes are reclassified from drone-only to truck-only, and the branch-and-cut is solved again.

*To speed up the procedure, a **time limit** is imposed for solving the reduced model.*



FS-TSP: ML-guided matheuristics

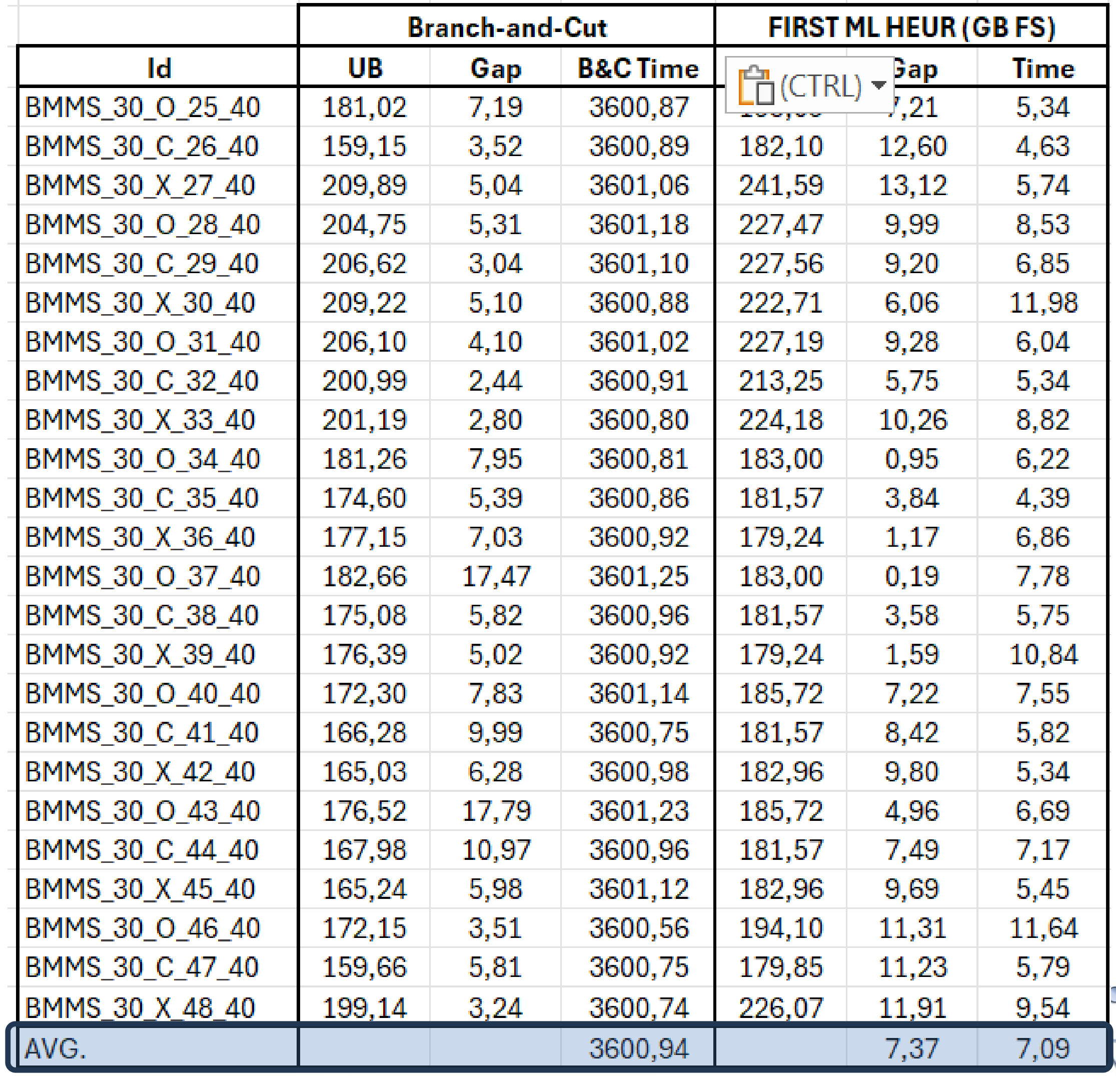
First matheuristic, computational results

Instances generated with Murray & Chu procedure with 30 customers - Dtl = 40

- *Branch-and-Cut vs. ML-guided matheuristic*
- *Branch-and-Cut time limit = one hour*
- *low computation time*
- *Significant gaps between heuristic and B&C solutions*

The gap shows how much the heuristic solution differs, in percentage terms, from the branch-and-cut one:

G



	Branch-and-Cut			FIRST ML HEUR (GB FS)		
Id	UB	Gap	B&C Time		Gap	Time
BMMS_30_O_25_40	181,02	7,19	3600,87	[illegible]	7,21	5,34
BMMS_30_C_26_40	159,15	3,52	3600,89	182,10	12,60	4,63
BMMS_30_X_27_40	209,89	5,04	3601,06	241,59	13,12	5,74
BMMS_30_O_28_40	204,75	5,31	3601,18	227,47	9,99	8,53
BMMS_30_C_29_40	206,62	3,04	3601,10	227,56	9,20	6,85
BMMS_30_X_30_40	209,22	5,10	3600,88	222,71	6,06	11,98
BMMS_30_O_31_40	206,10	4,10	3601,02	227,19	9,28	6,04
BMMS_30_C_32_40	200,99	2,44	3600,91	213,25	5,75	5,34
BMMS_30_X_33_40	201,19	2,80	3600,80	224,18	10,26	8,82
BMMS_30_O_34_40	181,26	7,95	3600,81	183,00	0,95	6,22
BMMS_30_C_35_40	174,60	5,39	3600,86	181,57	3,84	4,39
BMMS_30_X_36_40	177,15	7,03	3600,92	179,24	1,17	6,86
BMMS_30_O_37_40	182,66	17,47	3601,25	183,00	0,19	7,78
BMMS_30_C_38_40	175,08	5,82	3600,96	181,57	3,58	5,75
BMMS_30_X_39_40	176,39	5,02	3600,92	179,24	1,59	10,84
BMMS_30_O_40_40	172,30	7,83	3601,14	185,72	7,22	7,55
BMMS_30_C_41_40	166,28	9,99	3600,75	181,57	8,42	5,82
BMMS_30_X_42_40	165,03	6,28	3600,98	182,96	9,80	5,34
BMMS_30_O_43_40	176,52	17,79	3601,23	185,72	4,96	6,69
BMMS_30_C_44_40	167,98	10,97	3600,96	181,57	7,49	7,17
BMMS_30_X_45_40	165,24	5,98	3601,12	182,96	9,69	5,45
BMMS_30_O_46_40	172,15	3,51	3600,56	194,10	11,31	11,64
BMMS_30_C_47_40	159,66	5,81	3600,75	179,85	11,23	5,79
BMMS_30_X_48_40	199,14	3,24	3600,74	226,07	11,91	9,54
AVG.			3600,94		7,37	7,09

Optimization and Problem Solving Laboratory
Laboratorio di Ottimizzazione e Problem Solving

FS-TSP: ML-guided matheuristics

Key limitation of the described heuristic:

- the classifier is forced to assign every customer to either the truck or the drone, even when the probability of belonging to a given class is very low.

Second matheuristic:

Core Idea

- *classifiers can provide **probability scores** for class membership*
- *A Machine Learning classifier is used to assign customers to three categories:*

Truck only → *can only be served by the truck*

Drone only → *can only be served by the drone*

Not classified → *customers that cannot be clearly assigned to either class.*

* *Unclassified customers are those that the classifier cannot assign "truck-only" or "drone-only" with a probability above a given threshold.*

** *In our experiments, the threshold value was chosen so that 10% of the nodes remain unclassified.*

An increase in the complexity of the reduced problem is expected to lead to an improvement in the quality of the obtained solutions.

FS-TSP: ML-guided matheuristics

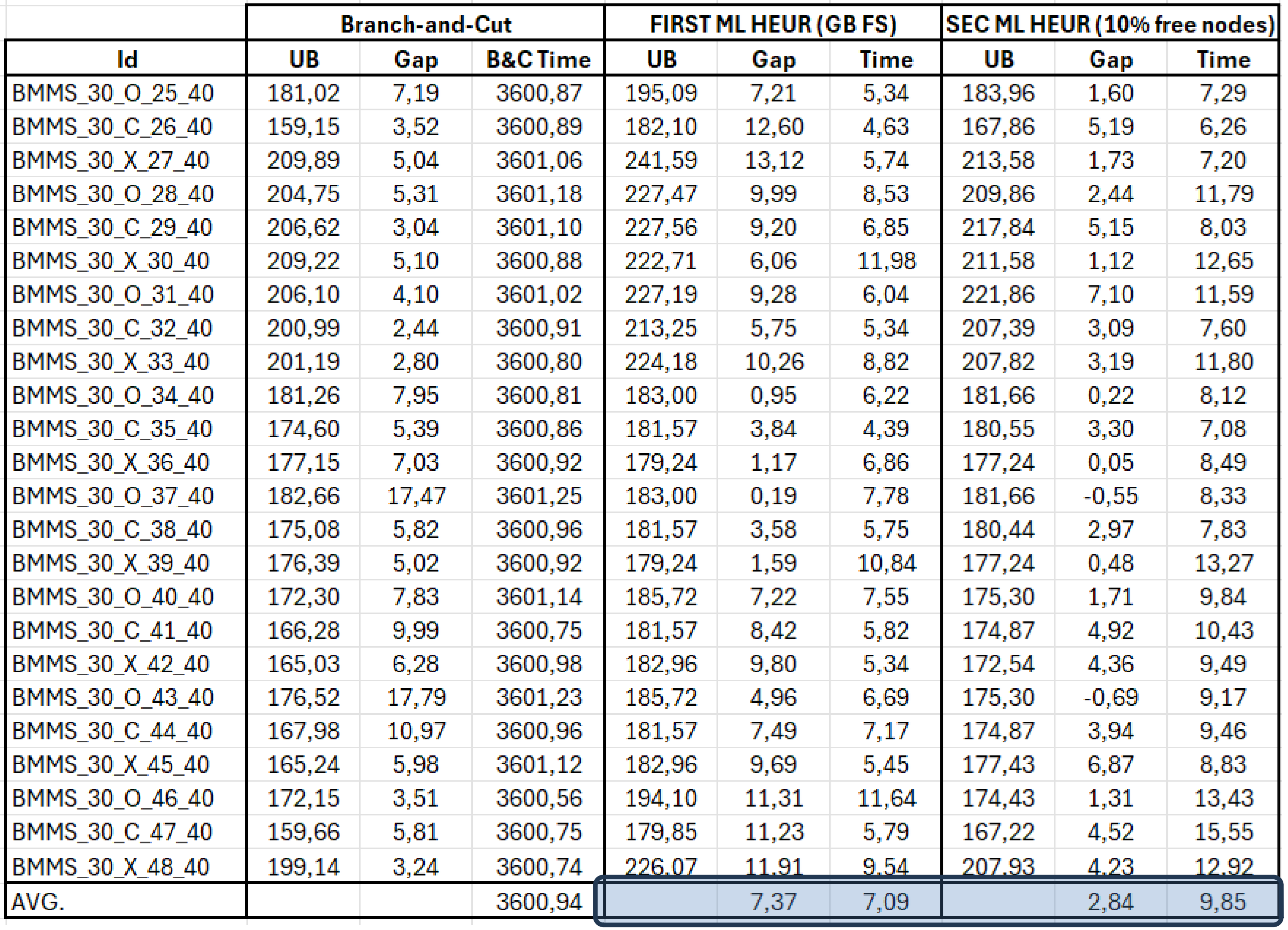
Second matheuristic, computational results

Instances generated with

Murray & Chu procedure

30 customers - Dtl = 40

G



	Branch-and-Cut			FIRST ML HEUR (GB FS)			SEC ML HEUR (10% free nodes)		
Id	UB	Gap	B&C Time	UB	Gap	Time	UB	Gap	Time
BMMS_30_O_25_40	181,02	7,19	3600,87	195,09	7,21	5,34	183,96	1,60	7,29
BMMS_30_C_26_40	159,15	3,52	3600,89	182,10	12,60	4,63	167,86	5,19	6,26
BMMS_30_X_27_40	209,89	5,04	3601,06	241,59	13,12	5,74	213,58	1,73	7,20
BMMS_30_O_28_40	204,75	5,31	3601,18	227,47	9,99	8,53	209,86	2,44	11,79
BMMS_30_C_29_40	206,62	3,04	3601,10	227,56	9,20	6,85	217,84	5,15	8,03
BMMS_30_X_30_40	209,22	5,10	3600,88	222,71	6,06	11,98	211,58	1,12	12,65
BMMS_30_O_31_40	206,10	4,10	3601,02	227,19	9,28	6,04	221,86	7,10	11,59
BMMS_30_C_32_40	200,99	2,44	3600,91	213,25	5,75	5,34	207,39	3,09	7,60
BMMS_30_X_33_40	201,19	2,80	3600,80	224,18	10,26	8,82	207,82	3,19	11,80
BMMS_30_O_34_40	181,26	7,95	3600,81	183,00	0,95	6,22	181,66	0,22	8,12
BMMS_30_C_35_40	174,60	5,39	3600,86	181,57	3,84	4,39	180,55	3,30	7,08
BMMS_30_X_36_40	177,15	7,03	3600,92	179,24	1,17	6,86	177,24	0,05	8,49
BMMS_30_O_37_40	182,66	17,47	3601,25	183,00	0,19	7,78	181,66	-0,55	8,33
BMMS_30_C_38_40	175,08	5,82	3600,96	181,57	3,58	5,75	180,44	2,97	7,83
BMMS_30_X_39_40	176,39	5,02	3600,92	179,24	1,59	10,84	177,24	0,48	13,27
BMMS_30_O_40_40	172,30	7,83	3601,14	185,72	7,22	7,55	175,30	1,71	9,84
BMMS_30_C_41_40	166,28	9,99	3600,75	181,57	8,42	5,82	174,87	4,92	10,43
BMMS_30_X_42_40	165,03	6,28	3600,98	182,96	9,80	5,34	172,54	4,36	9,49
BMMS_30_O_43_40	176,52	17,79	3601,23	185,72	4,96	6,69	175,30	-0,69	9,17
BMMS_30_C_44_40	167,98	10,97	3600,96	181,57	7,49	7,17	174,87	3,94	9,46
BMMS_30_X_45_40	165,24	5,98	3601,12	182,96	9,69	5,45	177,43	6,87	8,83
BMMS_30_O_46_40	172,15	3,51	3600,56	194,10	11,31	11,64	174,43	1,31	13,43
BMMS_30_C_47_40	159,66	5,81	3600,75	179,85	11,23	5,79	167,22	4,52	15,55
BMMS_30_X_48_40	199,14	3,24	3600,74	226,07	11,91	9,54	207,93	4,23	12,92
AVG.			3600,94		7,37	7,09		2,84	9,85

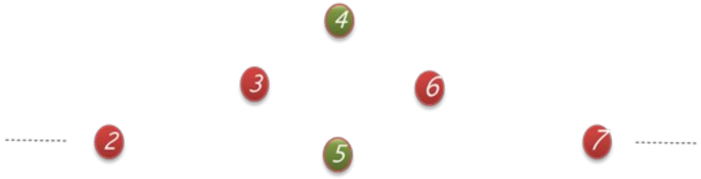
Optimization and Problem Solving Laboratory
Laboratorio di Ottimizzazione e Problem Solving

FS-TSP: ML-guided matheuristics

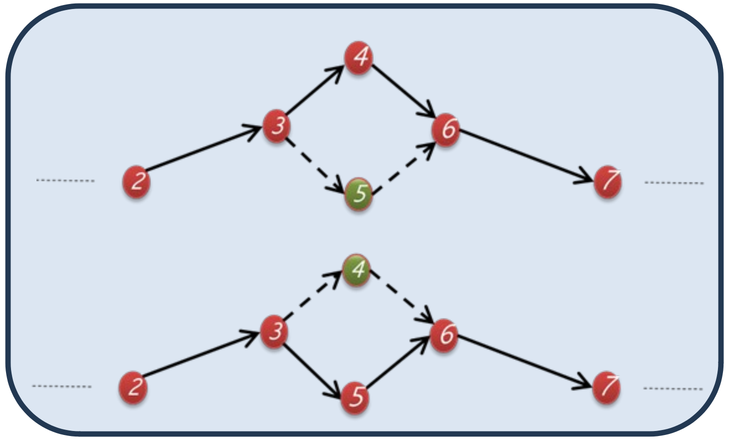
Further limitation of the second heuristic:

- Some customers are good drone candidates individually; but serving both with the drone leads to a low-quality solution.

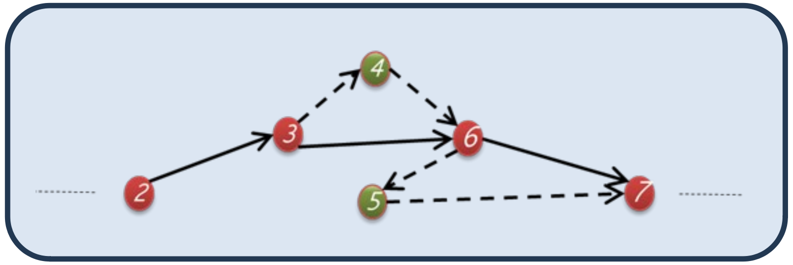
Example:



- *Nodes 4 and 5 are good candidates to be served by the drone*



- **Good solutions** → only node **4 or node 5** is served by the drone



- **Poor-quality solution** → both nodes **4 and 5 are** served by the drone

FS-TSP: ML-guided matheuristics

Further limitation of the second heuristic:

- Some customers are good drone candidates individually; but serving both with the drone leads to a low-quality solution.

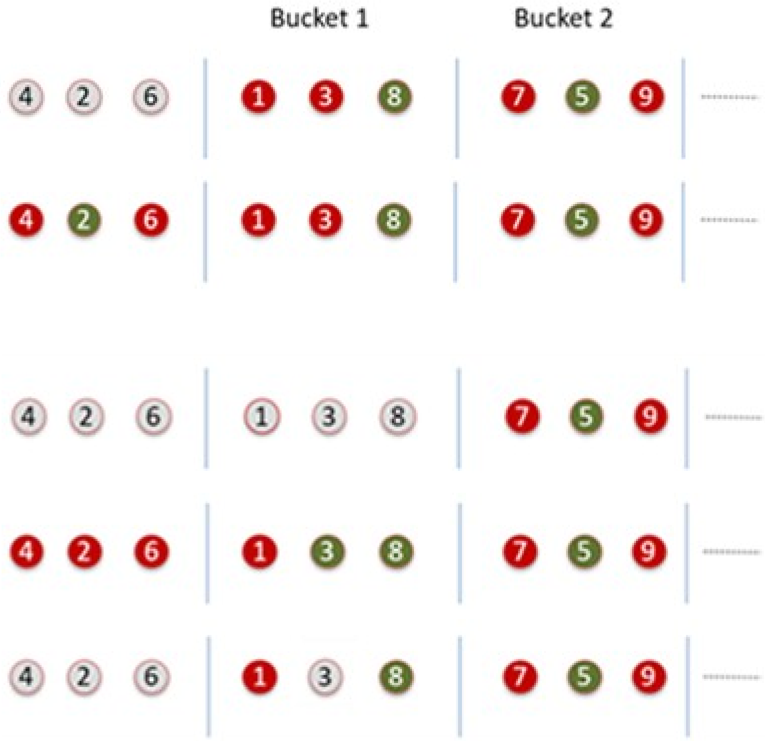
Third matheuristic (kernel search approach):

Kernel Search approach

- *Rank customers by their probability of belonging to the truck or drone class*
- *Define **kernel** as the set of unclassified nodes (low probability)*
- *Solve the reduced problem considering only the kernel nodes as unclassified*

- Iterative phase:

- *Add the **first unprocessed bucket** of nodes to the kernel*
- *Solve the reduced problem again*
- *If some bucket nodes change class, add them to the kernel*
- *Repeat with the unprocessed bucket until all are considered*



First reduced problem
unclussified nodes = kernel set

Initial solution

Second reduced problem
unclussified nodes = kernel + bucket 1

***New solution** – node 3 (outside the kernel) changed class*

New kernel set – node 3 added to the kernel

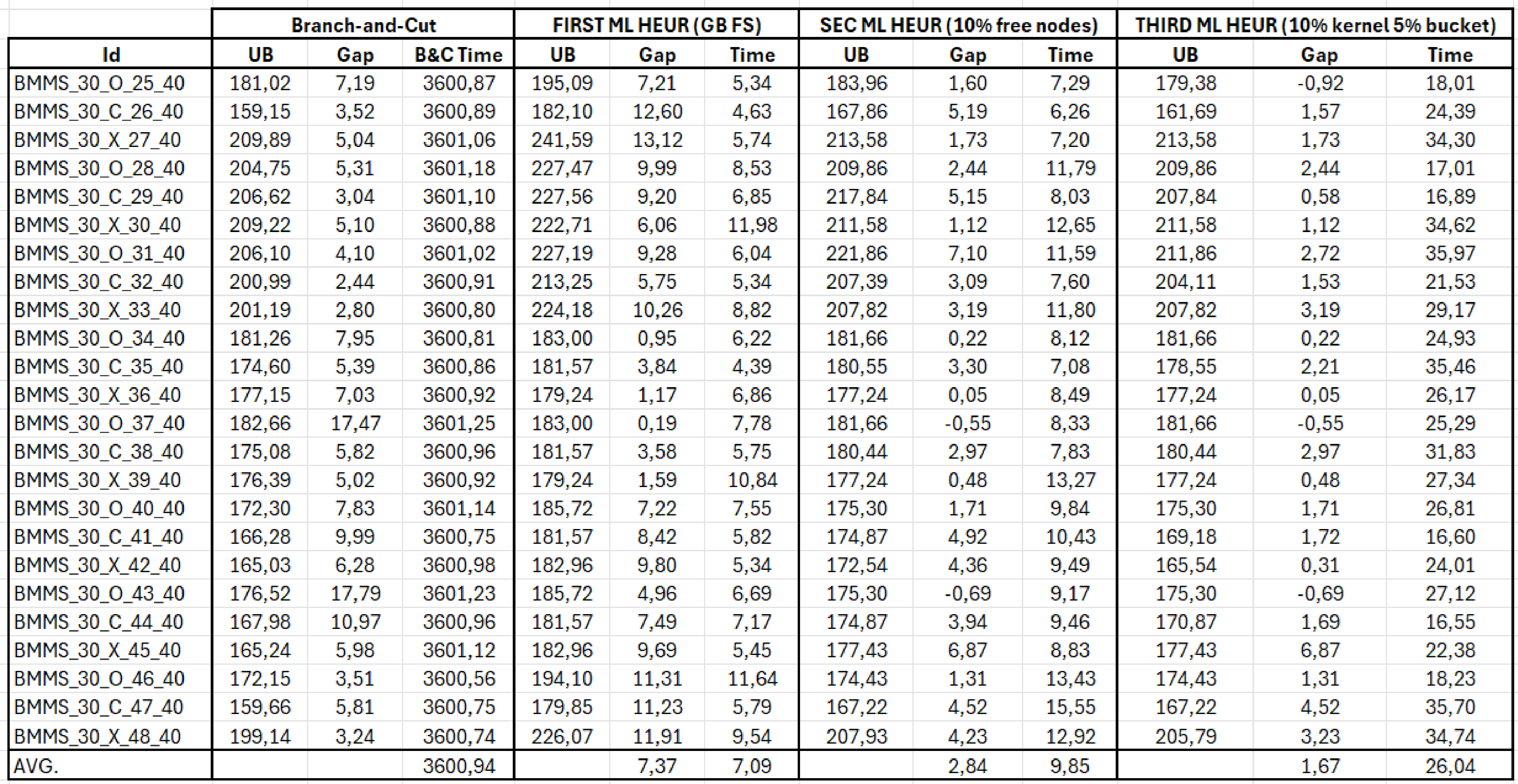
- Parameters:

*Initial kernel size = **10%** of nodes - Bucket size = **5%** of nodes - Time limit per iteration = **0.5 × number of nodes***

FS-TSP: ML-guided matheuristics

Third matheuristic, computational results

Instances generated with Murray & Chu procedure 30 customers - Dtl = 40



	Branch-and-Cut			FIRST ML HEUR (GB FS)			SEC ML HEUR (10% free nodes)			THIRD ML HEUR (10% kernel 5% bucket)		
Id	UB	Gap	B&C Time	UB	Gap	Time	UB	Gap	Time	UB	Gap	Time
BMMS_30_O_25_40	181,02	7,19	3600,87	195,09	7,21	5,34	183,96	1,60	7,29	179,38	-0,92	18,01
BMMS_30_C_26_40	159,15	3,52	3600,89	182,10	12,60	4,63	167,86	5,19	6,26	161,69	1,57	24,39
BMMS_30_X_27_40	209,89	5,04	3601,06	241,59	13,12	5,74	213,58	1,73	7,20	213,58	1,73	34,30
BMMS_30_O_28_40	204,75	5,31	3601,18	227,47	9,99	8,53	209,86	2,44	11,79	209,86	2,44	17,01
BMMS_30_C_29_40	206,62	3,04	3601,10	227,56	9,20	6,85	217,84	5,15	8,03	207,84	0,58	16,89
BMMS_30_X_30_40	209,22	5,10	3600,88	222,71	6,06	11,98	211,58	1,12	12,65	211,58	1,12	34,62
BMMS_30_O_31_40	206,10	4,10	3601,02	227,19	9,28	6,04	221,86	7,10	11,59	211,86	2,72	35,97
BMMS_30_C_32_40	200,99	2,44	3600,91	213,25	5,75	5,34	207,39	3,09	7,60	204,11	1,53	21,53
BMMS_30_X_33_40	201,19	2,80	3600,80	224,18	10,26	8,82	207,82	3,19	11,80	207,82	3,19	29,17
BMMS_30_O_34_40	181,26	7,95	3600,81	183,00	0,95	6,22	181,66	0,22	8,12	181,66	0,22	24,93
BMMS_30_C_35_40	174,60	5,39	3600,86	181,57	3,84	4,39	180,55	3,30	7,08	178,55	2,21	35,46
BMMS_30_X_36_40	177,15	7,03	3600,92	179,24	1,17	6,86	177,24	0,05	8,49	177,24	0,05	26,17
BMMS_30_O_37_40	182,66	17,47	3601,25	183,00	0,19	7,78	181,66	-0,55	8,33	181,66	-0,55	25,29
BMMS_30_C_38_40	175,08	5,82	3600,96	181,57	3,58	5,75	180,44	2,97	7,83	180,44	2,97	31,83
BMMS_30_X_39_40	176,39	5,02	3600,92	179,24	1,59	10,84	177,24	0,48	13,27	177,24	0,48	27,34
BMMS_30_O_40_40	172,30	7,83	3601,14	185,72	7,22	7,55	175,30	1,71	9,84	175,30	1,71	26,81
BMMS_30_C_41_40	166,28	9,99	3600,75	181,57	8,42	5,82	174,87	4,92	10,43	169,18	1,72	16,60
BMMS_30_X_42_40	165,03	6,28	3600,98	182,96	9,80	5,34	172,54	4,36	9,49	165,54	0,31	24,01
BMMS_30_O_43_40	176,52	17,79	3601,23	185,72	4,96	6,69	175,30	-0,69	9,17	175,30	-0,69	27,12
BMMS_30_C_44_40	167,98	10,97	3600,96	181,57	7,49	7,17	174,87	3,94	9,46	170,87	1,69	16,55
BMMS_30_X_45_40	165,24	5,98	3601,12	182,96	9,69	5,45	177,43	6,87	8,83	177,43	6,87	22,38
BMMS_30_O_46_40	172,15	3,51	3600,56	194,10	11,31	11,64	174,43	1,31	13,43	174,43	1,31	18,23
BMMS_30_C_47_40	159,66	5,81	3600,75	179,85	11,23	5,79	167,22	4,52	15,55	167,22	4,52	35,70
BMMS_30_X_48_40	199,14	3,24	3600,74	226,07	11,91	9,54	207,93	4,23	12,92	205,79	3,23	34,74
AVG.			3600,94		7,37	7,09		2,84	9,85		1,67	26,04

FS-TSP: ML-guided matheuristics

Third matheuristic, computational results

Heuristic approaches: current state of the art

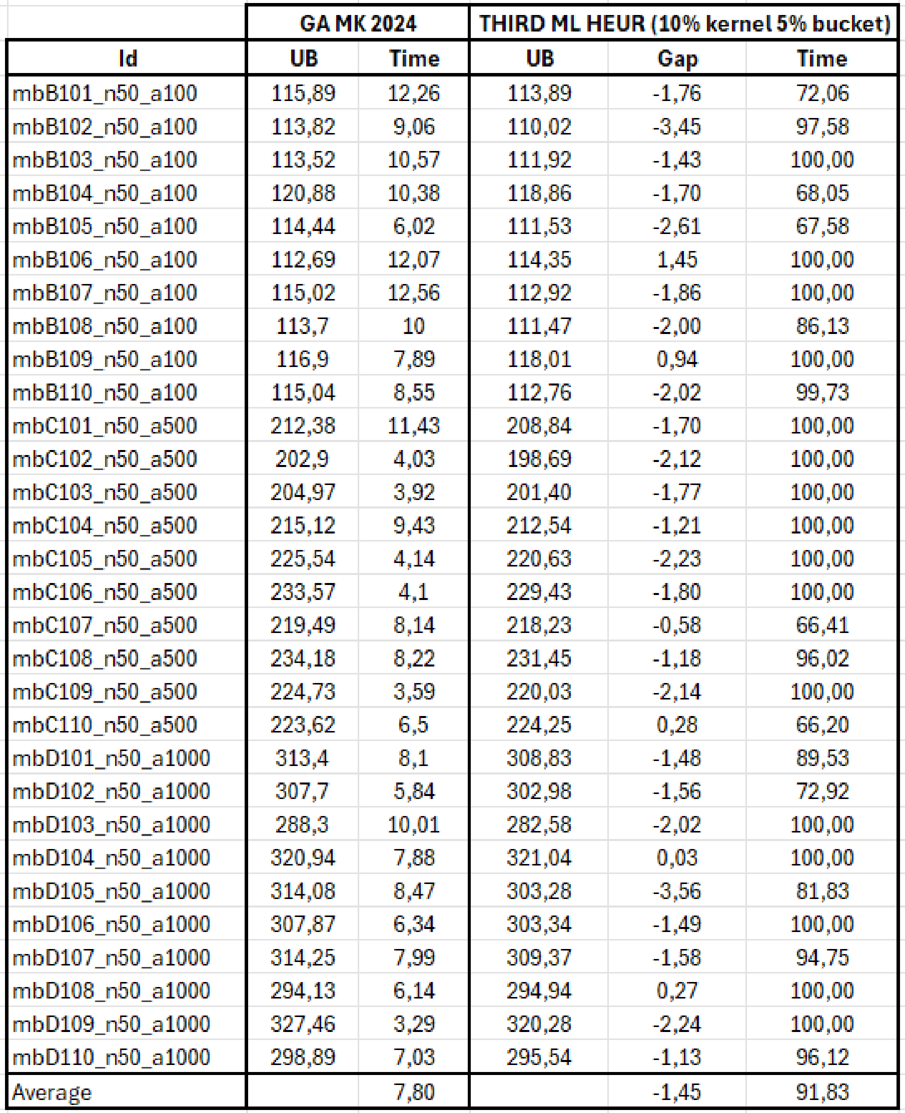
- **Sasan Mahmoudinazlou & Changhyun Kwon (2024)**
  A hybrid genetic algorithm with type-aware chromosomes for Traveling Salesman Problems with Drone
  Published in **EJOR**

Instances generated with Murray & Chu procedure

with 50 customers - Dtl = 20

- *Genetic algorithm vs. ML-guided matheuristic*
- *Mathheuristic time limit = 100 sec*
- *Maximum number of analyzed buckets = 5*
- *Beter solutions*
- *Significantly higher computational times*

G



	GA MK 2024		THIRD ML HEUR (10% kernel 5% bucket)		
Id	UB	Time	UB	Gap	Time
mbB101_n50_a100	115,89	12,26	113,89	-1,76	72,06
mbB102_n50_a100	113,82	9,06	110,02	-3,45	97,58
mbB103_n50_a100	113,52	10,57	111,92	-1,43	100,00
mbB104_n50_a100	120,88	10,38	118,86	-1,70	68,05
mbB105_n50_a100	114,44	6,02	111,53	-2,61	67,58
mbB106_n50_a100	112,69	12,07	114,35	1,45	100,00
mbB107_n50_a100	115,02	12,56	112,92	-1,86	100,00
mbB108_n50_a100	113,7	10	111,47	-2,00	86,13
mbB109_n50_a100	116,9	7,89	118,01	0,94	100,00
mbB110_n50_a100	115,04	8,55	112,76	-2,02	99,73
mbC101_n50_a500	212,38	11,43	208,84	-1,70	100,00
mbC102_n50_a500	202,9	4,03	198,69	-2,12	100,00
mbC103_n50_a500	204,97	3,92	201,40	-1,77	100,00
mbC104_n50_a500	215,12	9,43	212,54	-1,21	100,00
mbC105_n50_a500	225,54	4,14	220,63	-2,23	100,00
mbC106_n50_a500	233,57	4,1	229,43	-1,80	100,00
mbC107_n50_a500	219,49	8,14	218,23	-0,58	66,41
mbC108_n50_a500	234,18	8,22	231,45	-1,18	96,02
mbC109_n50_a500	224,73	3,59	220,03	-2,14	100,00
mbC110_n50_a500	223,62	6,5	224,25	0,28	66,20
mbD101_n50_a1000	313,4	8,1	308,83	-1,48	89,53
mbD102_n50_a1000	307,7	5,84	302,98	-1,56	72,92
mbD103_n50_a1000	288,3	10,01	282,58	-2,02	100,00
mbD104_n50_a1000	320,94	7,88	321,04	0,03	100,00
mbD105_n50_a1000	314,08	8,47	303,28	-3,56	81,83
mbD106_n50_a1000	307,87	6,34	303,34	-1,49	100,00
mbD107_n50_a1000	314,25	7,99	309,37	-1,58	94,75
mbD108_n50_a1000	294,13	6,14	294,94	0,27	100,00
mbD109_n50_a1000	327,46	3,29	320,28	-2,24	100,00
mbD110_n50_a1000	298,89	7,03	295,54	-1,13	96,12
Average		7,80		-1,45	91,83

OPS Lab
Optimization and Problem Solving Laboratory
Laboratorio di Ottimizzazione e Problem Solving

FS-TSP: ML-guided matheuristics

Third matheuristic, computational results

Heuristic approaches: current state of the art

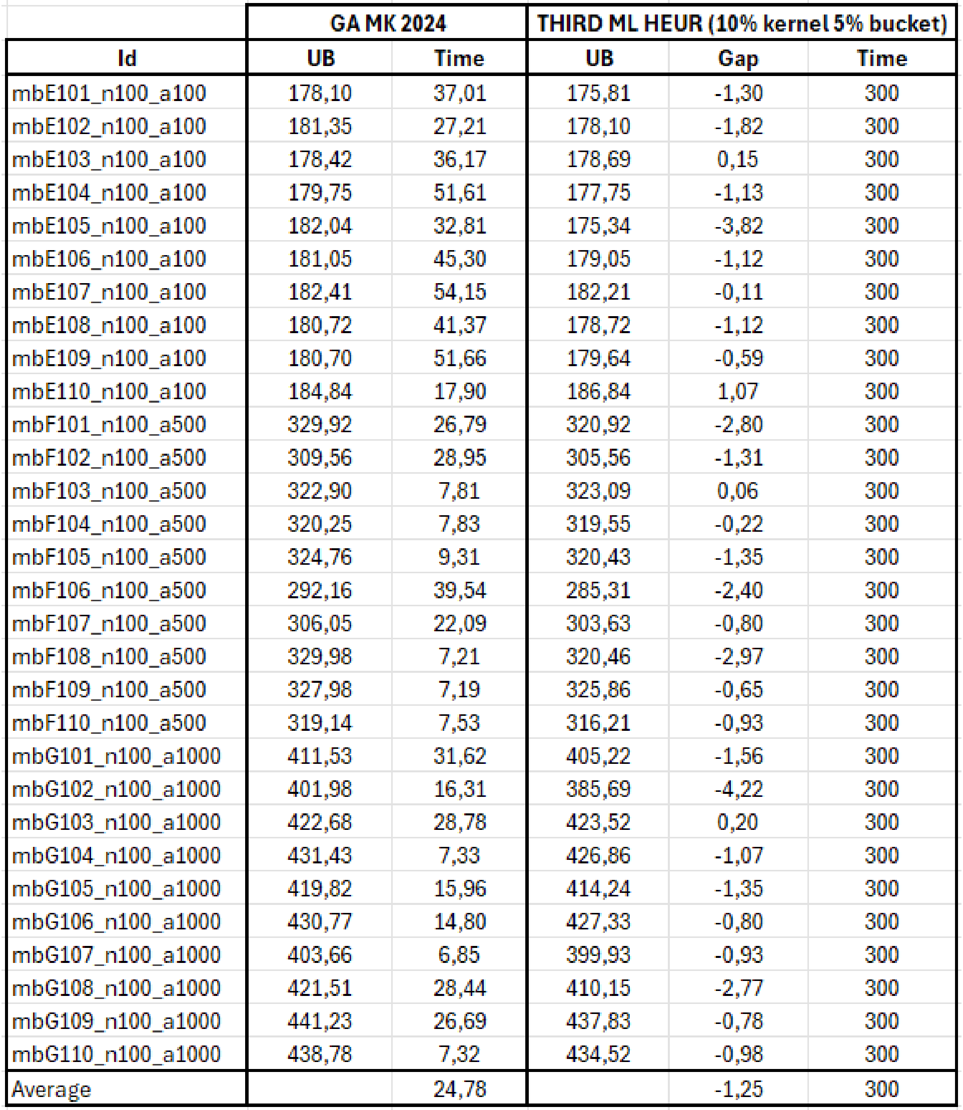
- **Sasan Mahmoudinazlou & Changhyun Kwon (2024)**
  A hybrid genetic algorithm with type-aware chromosomes for Traveling Salesman Problems with Drone
  Published in **EJOR**

Instances generated with Murray & Chu procedure

with 100 customers - Dtl = 20

- *Genetic algorithm vs. ML-guided matheuristic*
- *Mathheuristic time limit = 300 sec*
- *Maximum number of analyzed buckets = 5*
- *Beter solutions*
- *The matheuristic always runs until the time limit.*

G



	GA MK 2024		THIRD ML HEUR (10% kernel 5% bucket)		
Id	UB	Time	UB	Gap	Time
mbE101_n100_a100	178,10	37,01	175,81	-1,30	300
mbE102_n100_a100	181,35	27,21	178,10	-1,82	300
mbE103_n100_a100	178,42	36,17	178,69	0,15	300
mbE104_n100_a100	179,75	51,61	177,75	-1,13	300
mbE105_n100_a100	182,04	32,81	175,34	-3,82	300
mbE106_n100_a100	181,05	45,30	179,05	-1,12	300
mbE107_n100_a100	182,41	54,15	182,21	-0,11	300
mbE108_n100_a100	180,72	41,37	178,72	-1,12	300
mbE109_n100_a100	180,70	51,66	179,64	-0,59	300
mbE110_n100_a100	184,84	17,90	186,84	1,07	300
mbF101_n100_a500	329,92	26,79	320,92	-2,80	300
mbF102_n100_a500	309,56	28,95	305,56	-1,31	300
mbF103_n100_a500	322,90	7,81	323,09	0,06	300
mbF104_n100_a500	320,25	7,83	319,55	-0,22	300
mbF105_n100_a500	324,76	9,31	320,43	-1,35	300
mbF106_n100_a500	292,16	39,54	285,31	-2,40	300
mbF107_n100_a500	306,05	22,09	303,63	-0,80	300
mbF108_n100_a500	329,98	7,21	320,46	-2,97	300
mbF109_n100_a500	327,98	7,19	325,86	-0,65	300
mbF110_n100_a500	319,14	7,53	316,21	-0,93	300
mbG101_n100_a1000	411,53	31,62	405,22	-1,56	300
mbG102_n100_a1000	401,98	16,31	385,69	-4,22	300
mbG103_n100_a1000	422,68	28,78	423,52	0,20	300
mbG104_n100_a1000	431,43	7,33	426,86	-1,07	300
mbG105_n100_a1000	419,82	15,96	414,24	-1,35	300
mbG106_n100_a1000	430,77	14,80	427,33	-0,80	300
mbG107_n100_a1000	403,66	6,85	399,93	-0,93	300
mbG108_n100_a1000	421,51	28,44	410,15	-2,77	300
mbG109_n100_a1000	441,23	26,69	437,83	-0,78	300
mbG110_n100_a1000	438,78	7,32	434,52	-0,98	300
Average		24,78		-1,25	300

Conclusion and future work perspectives

Heuristic approaches

- *We presented an ML-guided matheuristic that achieves good results in terms of solution quality, but still requires high computational times.*
- *Future improvements include:*
  - ✓ *enhancing the solution of the reduced problem,*
  - ✓ *improving node classification quality,*
  - ✓ *training classifiers on different instance classes (e.g., Poikonen).*

Exact approaches

- *We described both a Branch-and-Cut method based on a Big-M-free MILP formulation and a Branch-and-Price approach based on ng-route relaxation.*
- *The latter is able to solve large-scale instances, although its performance deteriorates on certain instance classes.*
- *Future work should focus on improving their efficiency in order to handle more difficult instances, both in size and in topological complexity.*

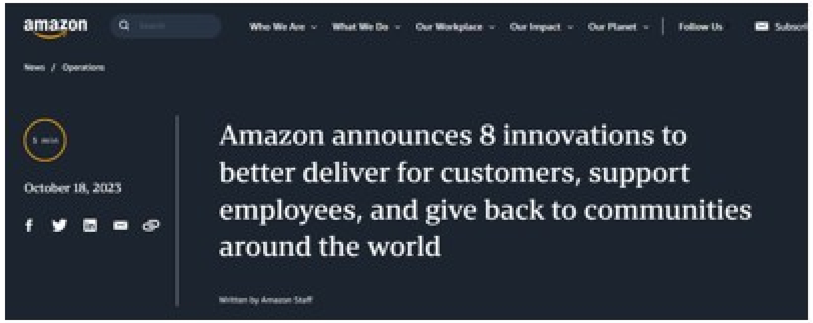
Conclusion and future work perspectives

Future Directions

- *Given the strong interest of both the scientific community and logistics operators, increasing attention will be devoted to exact and heuristic methods for the many variants of truck-and-drone problems*:
  - ✓ *multi-drone and multi-vehicle settings,*
  - ✓ *drones able to serve multiple customers per sortie,*
  - ✓ *integrated systems with drones, other autonomous vehicles, and parcel lockers.*
  - ✓ *….*

DELIVERY
The Rise of the Delivery Drones:
How Wing Plans to Deliver Millions
of Packages by 2024

ISP TODAY



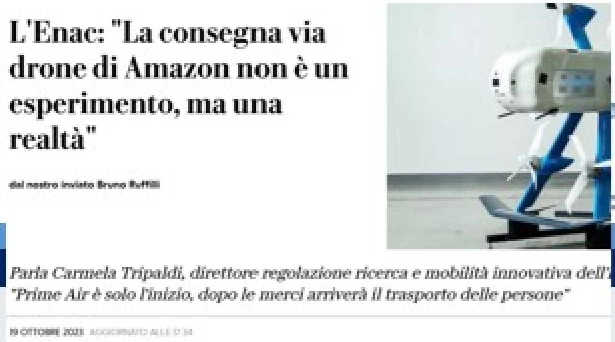
News / Operations

October 18, 2023

Amazon announces 8 innovations to better deliver for customers, support employees, and give back to communities around the world

Written by Amazon Staff

At last year's Delivering the Future event, we announced the prototype of our latest drone design, the MK30. This year, we revealed a first look at the MK30, which will launch in 2024.


L'Enac: "La consegna via drone di Amazon non è un esperimento, ma una realtà"

dal nostro inviato Bruno Ruffilli

Parla Carmela Tripaldi, direttore regolazione ricerca e mobilità innovativa dell'
"Prime Air è solo l'inizio, dopo le merci arriverà il trasporto delle persone"

19 OTTOBRE 2023 AGGIORNATO ALLE 17:34

Thank you for your attention!

…hoping no one is following her!

OPS Lab

Optimization and Problem Solving Laboratory
Laboratorio di Ottimizzazione e Problem Solving