A GRASP metaheuristic for the last-mile Vehicle Routing Problem with Delivery Options

ELEFTHERIOS G. MANOUSAKIS[1]

IOANNIS F. KITSOS KALYVIANAKIS[1]

ANGELOS K. OMIROLIS[1]

EMMANOUIL E. ZACHARIADIS[1]

[1] DEPARTMENT OF MANAGEMENT SCIENCE & TECHNOLOGY, ATHENS UNIVERSITY OF ECONOMICS AND BUSINESS

33rd EURO, 30th June – 3rd July 2024, Technical University of Denmark (DTU), Copenhagen, Denmark

Presentation Outline

Introduction

- The last-mile challenge
- Literature

The Vehicle Routing Problem with Delivery Options

- Visual example
- Definition

Solution methodology

- Initial solution construction
- Local search

Computational Results

Conclusions

Introduction

THE LAST-MILE CHALLENGE

Last-mile delivery

The final step of the delivery process where goods are transported from a transportation hub to their final destination (e.g., customer's home).

There has never been a time of greater demand for last-mile transport

- Last mile market size (Global Market Insights, GMI 2024)
  - 2023: 175.3$ billions
  - 2032: 305.4$ billions
  - North America contributes the 37% of this (2023).
  - Increasing trend in many markets around the globe

Last mile is the costliest link in the supply chain

- 41% of overall supply chain costs (almost double of all other processes, i.e., parceling, warehousing) (Cemex Ventures, 2023).

Challenges

Urban Congestion

Freight traffic contributes to 20% of urban traffic congestion

Environmental Impact

Freight transport accounts for approximately 25% of greenhouse gas emissions

High Delivery Costs

The net profit margins for many transport companies are minimal, often negligible

Customer Expectations

There is an increasing demand for faster, more flexible, and reliable delivery options, including shorter delivery windows, accurate time predictions, and same-day deliveries.

Optimization

Operations Research: numerous models to optimize cases of delivery processes:

- enhancing resources efficiency
- minimizing costs
- increasing customer satisfaction

The foundational VRP model lies below several specialized case specific realistic variants:

- CVRP (Cumulative VRP): Manages accumulated cost (e.g., for satisfying latest arrival)
- VRPTW (VRP with Time Windows): Incorporates specific delivery time frames
- VRPPD (VRP with Pickup and Delivery): Handles both delivery and pickup tasks

Vehicle Routing Problem with Delivery Options

Seminal papers:

- Tilk, C., Olkis, K. and Irnich, S. (2021), “The last-mile vehicle routing problem with delivery options”, *OR Spectrum*, Springer Berlin Heidelberg, Vol. 43 No. 4, pp. 877–904.
- Dumez, D., Lehuédé, F. and Péton, O. (2021), “A large neighborhood search approach to the vehicle routing problem with delivery options”, Transportation Research Part B: Methodological, Elsevier Ltd, Vol. 144, pp. 103–132.

Motivated by last mile delivery challenges: “the bottleneck of e-commerce” (Wang et al. 2014) & “the logistic service providers’ nightmare(s)” (Savelsbergh and Van Woensel 2016).

Extends

- Vehicle Routing Problem with Time Windows (VRPTW)
- Generalized Vehicle Routing Problem (GVRP)

Vehicle Routing Problem with Delivery Options

Innovation

- Alternative customer delivery options with ranking
- Capacitated shared facilities

Challenges

- Time windows
- Synchronized resources (shared locations capacity, priorities)
- New structure of the search space, due to the presence of alternative delivery locations for each customer

Literature & Motivation

Cardeneo (2005)

- Introduced the initial basic version of the Vehicle Routing Problem (VRP) with alternative delivery locations

Los et al. (2018)

- Considered service levels and customer preferences, along with location selection, in the generalized pickup and delivery problem with time windows and preferences

Ozbaygin et al. (2017); Reyes et al. (2017)

- Addressed the vehicle routing problem with home and roaming delivery locations (VRP(H)RDL), a special case of the VRPDO

Lombard et al. (2018)

- Explored the VRP(H)RDL with stochastic travel times

Tilk et al. (2021): Introduced the VRPDO

- branch-and-price algorithm featuring two different network structures, cutting planes, and branching rules
- state-of-the-art algorithm on benchmark instances for VRPHRDL and VRPRDL

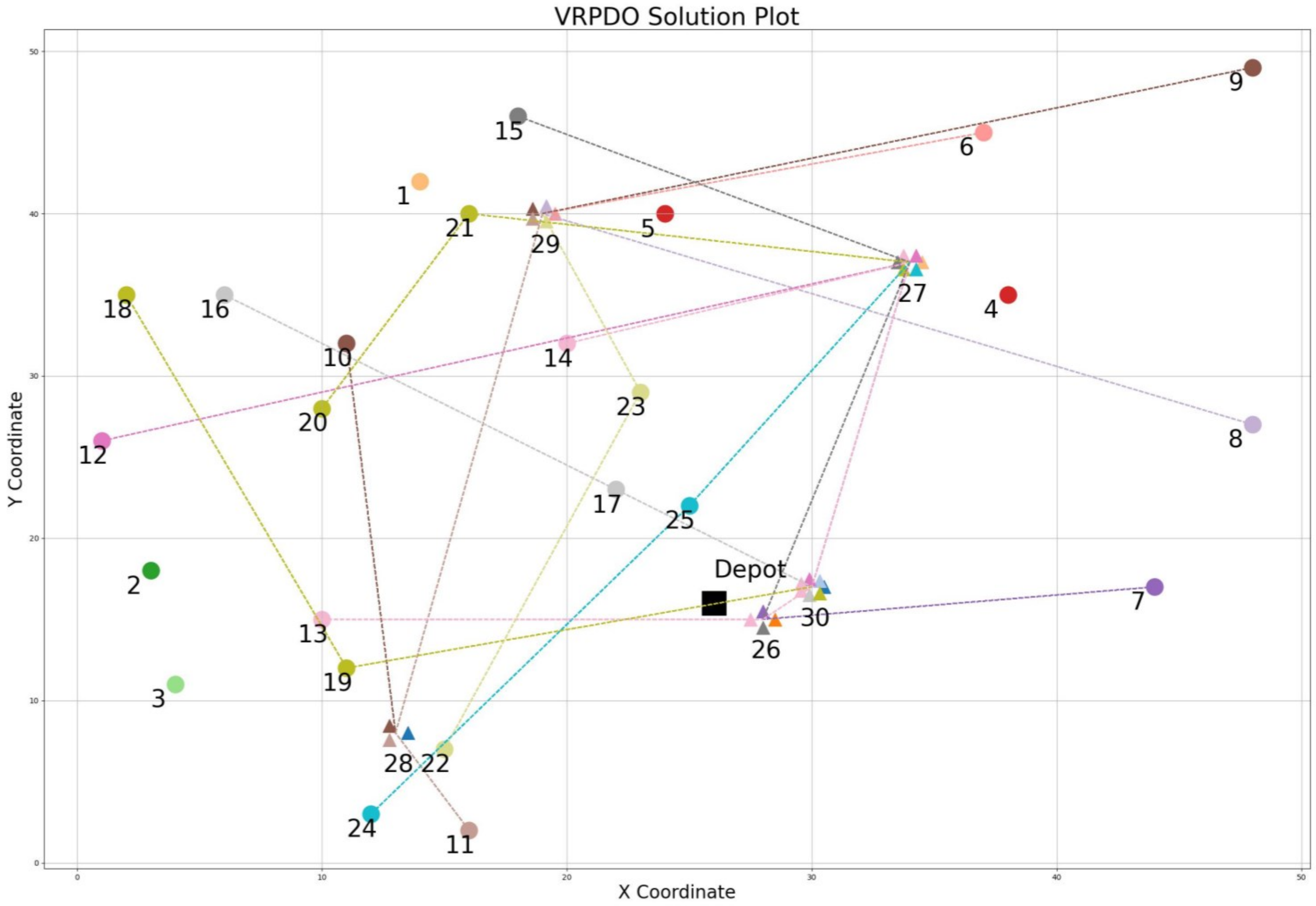
Dumez et al. (2021): Introduced the VRPDO

- Large neighborhood search algorithm with ruin and recreate operators
- A set partitioning problem is periodically used to reassemble routes

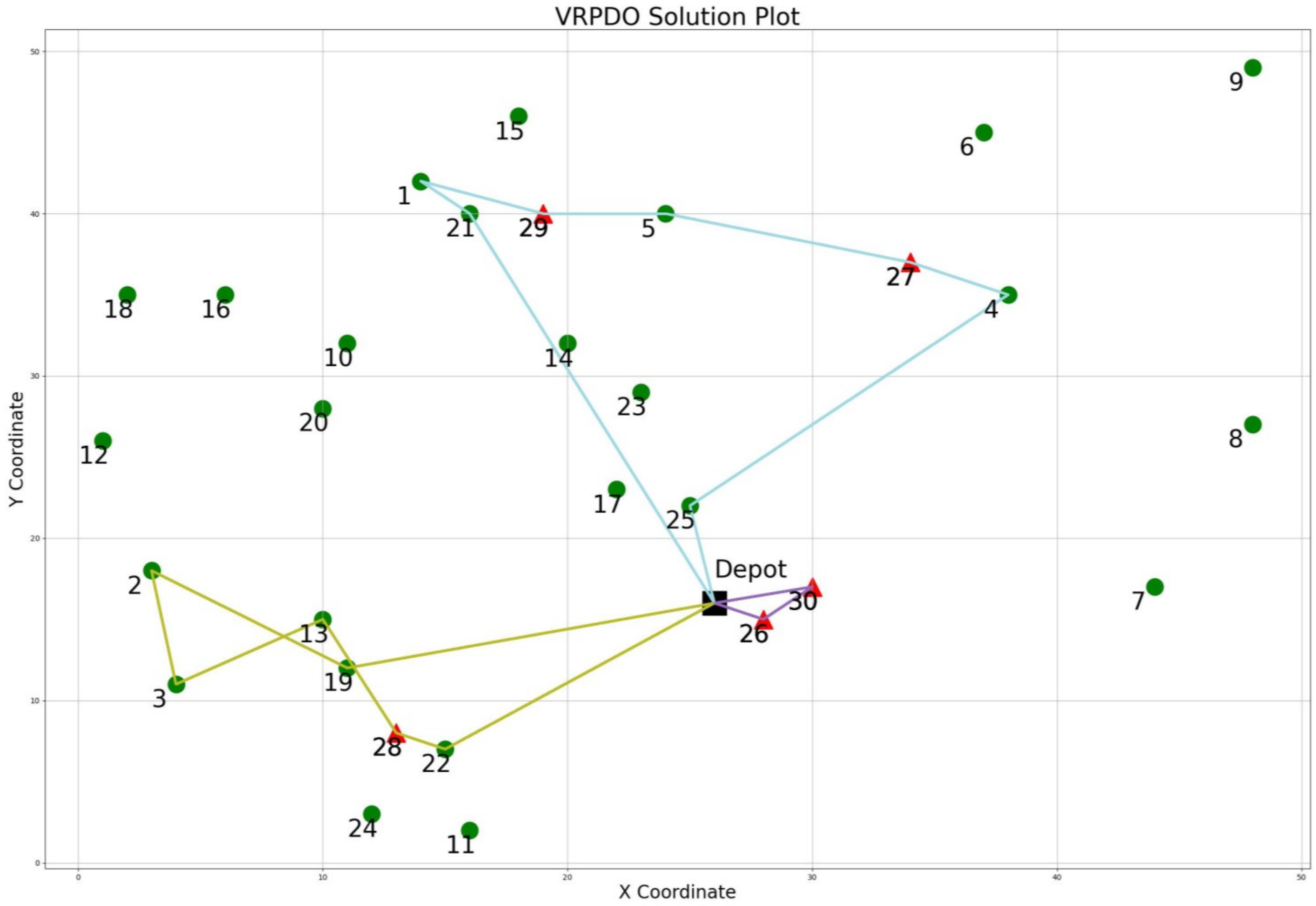
Problem definition

VISUAL EXAMPLE AND MODEL

Plotting an instance



Plotting the solution



Vehicle Routing Problem with Delivery Options

Objective:

Minimizing the number of **vehicles** and the total travel **cost** for serving all customers.

Constraints:

Limited number of capacitated **vehicles**

Time windows

All customer must be served

Shared **locations capacity**

Service levels to satisfy priorities

Decisions:

Which option to chose for each customer?

Which are the **best routes** to serve the selected options?

The model is explained in detail in Dumez et al. (2021).

Methodology

GRASP METAHEURISTIC

Grasp metaheuristic algorithm

Iteratively follow construct initial solutions and improve via local search

Step 1: Construct initial solutions

- Minimum insertion algorithm
- Solution pool for diversified option combinations

Step 2: Iteratively improve the solutions via Local search

- Scheme design
- Promises to avoid cycling
- Routing and option operators/neighborhoods

Approach 1 – Minimum insertion

- **Minimum insertion**: Use minimum insertion to find the best three insertions in each loop until all customer are served.
- **Vehicle number minimization**: Penalize insertions in empty or near empty vehicles to ensure least number of vehicles while maintaining feasibility.
- **RCL (Restricted Candidate List)**: Randomly select one of the top three solutions for diversified, high-quality restarts.

Algorithm 1 Overall Scheme - Minimum insertion

```
S ← MinimumInsertionAlgorithm(), S* ← ∅
for i ← 1 to restarts do
    S_i* ← LocalSearch(S_i)
    if Z(S_i*) > Z(S*) then
        S* ← S_i*
    end if
end for
return S*
```

Approach 1 – Minimum insertion

- **Minimum insertion**: Use minimum insertion to find the best three insertions in each loop until all customer are served.
- **Vehicle number minimization**: Penalize insertions in empty or near empty vehicles to ensure least number of vehicles while maintaining feasibility.
- **RCL (Restricted Candidate List):** Randomly select one of the top three solutions for diversified, high-quality restarts.

Algorithm 1 Overall Scheme - Minimum insertion

$S \leftarrow \mathit{MinimumInsertionAlgorithm}()$, $S^* \leftarrow \emptyset$
for $i \leftarrow 1$ to *restarts* **do**
  $S_i^* \leftarrow \mathit{LocalSearch}(S_i)$
  if $Z(S_i^*) > Z(S^*)$ **then**
    $S^* \leftarrow S_i^*$
  end if
end for
return S^*

Problem: Low diversity in options and eachs eac replacement alters the network.

Approach 2 – Solution pool

Predetermined Options:

- Use a weighted metric to select options for each customer:
  - Distance from closest nodes
  - Compatibility of time windows with neighbors
  - Time windows span

Route Construction:

- Apply the minimum insertion algorithm to route all preselected options.
- Repeat until a satisfactory number of solutions is available.

Approach 2 – Solution pool

Predetermined Options:

- Use a weighted metric to select options for each customer:
  - Distance from closest nodes
  - Compatibility of time windows with neighbors
  - Time windows span

Route Construction:

- Apply the minimum insertion algorithm to route all preselected options.
- Repeat until a satisfactory number of solutions is available.

Solution Pool:

- **Advantage:** Focus on the problem network affects the search space.
- **Disadvantage:** Manual weighting; a learning procedure is worth investigating.

Approach 2 – Solution pool

Algorithm 2 Overall Scheme - Solution pool

$S^* \leftarrow \emptyset$
$O^{selected} \leftarrow FindBestOptions()$
$S \leftarrow MinimumInsertionAlgorithm(O^{selected})$
for $i \leftarrow 1$ to $restarts$ **do**
    $S_i^* \leftarrow LocalSearch(S_i)$
    if $Z(S_i^*) > Z(S^*)$ **then**
        $S^* \leftarrow S_i^*$
    end if
end for
return S^*

Local search scheme

Multiple Restarts:

- Set maximum iterations and limit non-improving iterations.

Move Filtering/Tabu Policy:

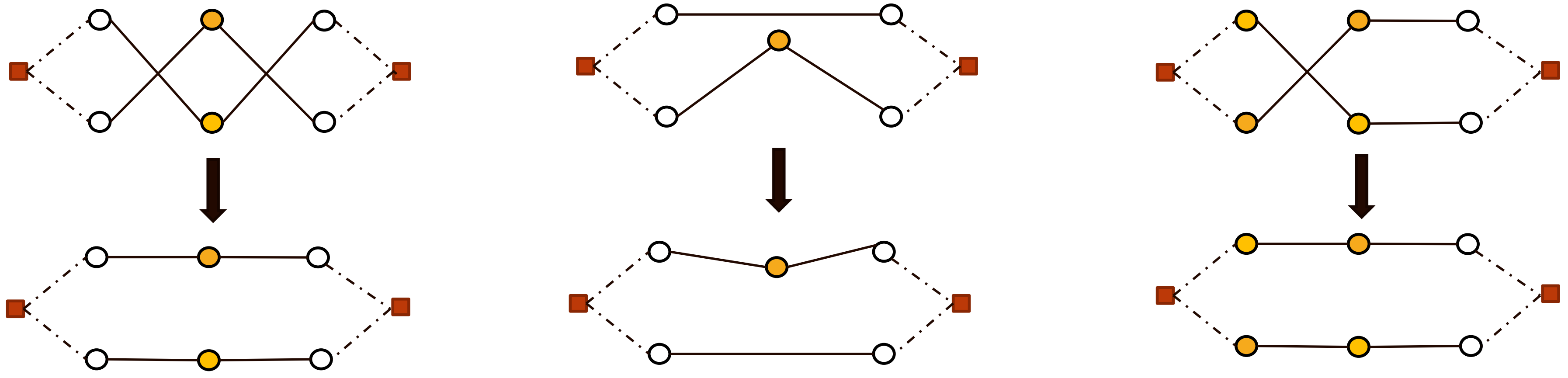
- Utilize the promise mechanism by Zachariadis et al. (2015).

Neighborhood Exploration:

- Explore all neighborhoods in each iteration using five operators:
  - three classic routing operators
  - two option-related operators, controlled due to network alteration and combinatorial impact.

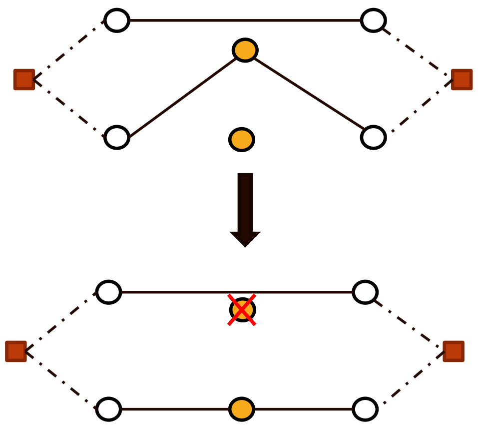
Classic routing operators

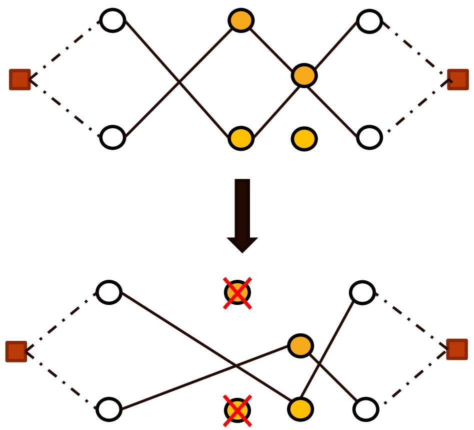
1. **Swap**: exchanges the positions of two selected options in the same or different routes.
2. **Relocation**: moves a single option from its current position to another position within the same route or to a different route.
3. **2-Opt**: remove two edges from the same or different routes and reconnect the two resulting paths in a different way to form a new route



Option-related operators

1. **Flip**: replaces one option with another option of the same customer (different location)
2. **Priority swap**: exchanges the positions of two selected options in the same or different routes replacing both option with other options of the same customer





Computational Results

BENCHMARKING AND EXPERIMENTS

Computational Experiments

- Benchmarking against 120 instances of Tilk et al. 2021:
  - Requests: 25 or 50
  - Classes: V (~1.5 options per request), U (~2 options per request). Priorities between 1 and 3 are uniformly distributed over the options of a request
  - Time windows: small (60-240 min), medium (120-480 min), large (240-600).
  - Location preparation time (e.g. parking): 6 min individual location, 4 min for shared

- Performance of BPC (Tilk et al. 2021)
  - VRPDO: 78 of the 120 instances solved to optimality.
  - VRPRDL: 17 new optimal solutions (x20 times faster than the former state of the art)

Computational Experiments

Implementation:

- Language: C# (.Net 6.0) with Visual Studio
- Machine: AMD Threadripper PRO 5955WX (16 cores, 4001 MHz), 128 GB RAM, x64 Windows 11

Settings & Parameters:

- Minimum insertions construction algorithm
- 10 restarts
- Each restart ends after 5,000 non-improving iterations or 15,000 total iterations
- Promises restart after 1.5 times the option set size

Benchmarking

			Tilk et al. 2021				Our algorithm			Comparison	
Class	Customers	Time windows	Optimal	Avg routes	Avg cost	Avg time	Avg routes	Avg cost	Avg time	# New best (Opt)	Gap (%)
U	25	S	10	3.00	2455.80	23.56	3.00	2651.80	165.18	0 (0)	8.13
		M	9	3.10	2192.30	1089.33	3.00	2207.00	345.76	3 (2)	0.77
		L	9	3.00	2440.60	1595.00	3.00	2480.90	418.29	2 (1)	1.73
	50	S	6	5.88	3821.75	4020.63	5.30	4149.55	870.96	3 (0)	6.07
		M	1	5.67	4286.89	7014.24	5.60	4202.10	1598.74	6 (0)	0.93
		L	1	5.40	3807.20	6559.36	5.50	4142.10	1690.21	5 (0)	6.05
V	25	S	10	3.00	2616.90	33.94	3.00	2694.20	244.34	1 (1)	2.92
		M	10	3.00	2443.60	489.54	3.00	2492.70	333.98	3 (3)	2.09
		L	10	3.00	2114.70	938.31	3.00	2120.80	531.77	8 (8)	0.25
	50	S	6	5.70	4392.20	4090.85	5.90	4465.40	514.29	2 (0)	1.73
		M	4	5.38	3722.63	4861.73	5.50	3939.70	1931.95	2 (0)	2.89
		L	2	5.71	3816.86	6227.61	5.50	3854.20	2186.02	4 (0)	-0.35
			78	**4.32**	**3175.95**	**3078.68**	**4.28**	**3283.37**	**902.62**	**39 (15)**	**2.77**

Benchmarking

Observations
- Marginally less vehicles
- 24 new best & 15 optimal
- Faster on average

			Tilk et al. 2021				Our algorithm			Comparison	
Class	Customers	Time windows	Optimal	Avg routes	Avg cost	Avg time	Avg routes	Avg cost	Avg time	# New best (Opt)	Gap (%)
U	25	S	10	3.00	2455.80	23.56	3.00	2651.80	165.18	0 (0)	8.13
		M	9	3.10	2192.30	1089.33	3.00	2207.00	345.76	3 (2)	0.77
		L	9	3.00	2440.60	1595.00	3.00	2480.90	418.29	2 (1)	1.73
	50	S	6	5.88	3821.75	4020.63	5.30	4149.55	870.96	3 (0)	6.07
		M	1	5.67	4286.89	7014.24	5.60	4202.10	1598.74	6 (0)	0.93
		L	1	5.40	3807.20	6559.36	5.50	4142.10	1690.21	5 (0)	6.05
V	25	S	10	3.00	2616.90	33.94	3.00	2694.20	244.34	1 (1)	2.92
		M	10	3.00	2443.60	489.54	3.00	2492.70	333.98	3 (3)	2.09
		L	10	3.00	2114.70	938.31	3.00	2120.80	531.77	8 (8)	0.25
	50	S	6	5.70	4392.20	4090.85	5.90	4465.40	514.29	2 (0)	1.73
		M	4	5.38	3722.63	4861.73	5.50	3939.70	1931.95	2 (0)	2.89
		L	2	5.71	3816.86	6227.61	5.50	3854.20	2186.02	4 (0)	-0.35
			78	**4.32**	**3175.95**	**3078.68**	**4.28**	**3283.37**	**902.62**	**39 (15)**	**2.77**

Conclusion

KEY FINDINGS & FUTURE WORK

Conclusion

Highlights

GRASP metaheuristic for VRPDO

Alternative construction heuristics (pool)

24/120 new best solutions

15/120 proven optimal

Future Work

Solve Dumez et al. (2021) large instances (50-400)

Inject mathematical programming/constraint programming components into the scheme

Machine learning model for option selecting

Bibliography

Tilk, C., Olkis, K. and Irnich, S. (2021), "The last-mile vehicle routing problem with delivery options", OR Spectrum, Springer Berlin Heidelberg, Vol. 43 No. 4, pp. 877–904.

Cardeneo A, (2005). Modellierung und Optimierung des B2C-Tourenplanungsproblems mit alternativen Lieferorten und -zeiten: Zugl.: Karlsruhe, Univ., Diss., (2005) volume 66 of Wissenschaftliche Ber- ichte des Institutes für Fördertechnik und Logistiksysteme der Universität Karlsruhe (TH). Univer- sitätsverlag, Karlsruhe, Germany (in German)

Los J, Spaan MTJ, Negenborn RR (2018) Fleet management for pickup and delivery problems with mul- tiple locations and preferences. In: Freitag M, Kotzab H, Pannek J (eds) Dynamics in logistics, vol 61. Lecture notes in logistics. Springer, Cham, pp 86–94

Ozbaygin G, Ekin Karasan O, Savelsbergh M, Yaman H (2017) A branch-and-price algorithm for the vehicle routing problem with roaming delivery locations. Transp Res Part B: Methodol 100:115–137

Lombard A, Tamayo-Giraldo S, Fontane F (2018) Vehicle routing problem with roaming delivery loca- tions and stochastic travel times (VRPRDL-S). Transp Res Procedia 30:167–177

Wang X, Zhan L, Ruan J, Zhang J (2014) How to choose last mile delivery modes for e-fulfillment. Math Probl Eng 2014:1–11

Savelsbergh M, Van Woensel T (2016) 50th anniversary invited article–city logistics: challenges and opportunities. Transp Sci 50(2):579–590

Global Market Insights, GMI (2024) Last Mile Delivery Market, https://www.gminsights.com/industry-analysis/last-mile-delivery-market

CemexVentures (2023) Last mile delivery: The logistic challenge, https://www.cemexventures.com/what-is-last-mile-delivery-2/

Dumez, D., Lehuédé, F. and Péton, O. (2021), "A large neighborhood search approach to the vehicle routing problem with delivery options", Transportation Research Part B: Methodological, Elsevier Ltd, Vol. 144, pp. 103–132.

Acknowledgements

H.F.R.I.
Hellenic Foundation for Research & Innovation

The research project was supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.) under the "2nd Call for H.F.R.I. Research Projects to support Faculty Members & Researchers" (Project Number: 02562).

Thank You!