



HAL
open science

Vehicle Routing Problem with Fair Profits and Time Windows

Aitor López Sánchez, Marin Lujak, Frédéric Semet, Holger Billhardt

► **To cite this version:**

Aitor López Sánchez, Marin Lujak, Frédéric Semet, Holger Billhardt. Vehicle Routing Problem with Fair Profits and Time Windows. IEEE Conference on Systems, Man, and Cybernetics, IEEE, Oct 2023, Honolulu, United States. hal-04395379

HAL Id: hal-04395379

<https://inria.hal.science/hal-04395379>

Submitted on 15 Jan 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Vehicle Routing Problem with Fair Profits and Time Windows

Aitor López Sánchez

CETINIA, University Rey Juan Carlos, Spain
Univ. Lille, CNRS, Inria, Centrale Lille, UMR 9189 CRIStAL
F-59000 Lille, France

Marin Lujak

CETINIA, University Rey Juan Carlos, Spain

Frédéric Semet

Univ. Lille, CNRS, Inria, Centrale Lille, UMR 9189 CRIStAL
F-59000 Lille, France

Holger Billhardt

CETINIA, University Rey Juan Carlos, Spain

June 2023

Abstract

In crowdsourced delivery organizations, where individual vehicles with shared common goals may have conflicting individual interests, the preference is for collaboration over competition, provided it is less costly. However, achieving a balance between the efficiency of individual vehicles and the overall fleet poses a challenge. This paper introduces a novel Vehicle Routing Problem with Fair Profits and Time Windows (VRP-FPTW), which aims to meet customer demand and stringent time windows while maximizing the profit of the worst-off vehicle in the fleet. We propose a centralized and distributed vehicle routing model for this problem, both with quality of solution guarantees. The distributed approach is tailored for multi-agent systems relying on a coordination mechanism where each vehicle modeled as an individually rational agent finds its route autonomously in coordination with a fleet coordinator agent, without sharing its private information. The objective of a vehicle agent is to maximize its own profit while following the fleet's norms and regulations based on shared values. Simulation experiments provide compelling evidence of the robustness and scalability of the proposed distributed approach, showcasing significant enhancements in both solution quality and computational efficiency, particularly when dealing with larger vehicle fleets.

Keywords: multi-agent system, distributed computing, column generation, vehicle routing problem, fairness, profit sharing

1 Introduction

In crowdsourced delivery fleets where vehicle drivers are individually rational and mutually competitive, collaboration is preferable if it is less costly than the competition. In this setting, the vehicles share common goals while their individual interests may be mutually conflicting. Each vehicle’s objective is to maximize its profit while obeying the norms and regulations of the fleet in line with its value system. These arise from the values statement that reflects the organization’s core principles and ethics code. Applications, among others, relate to last-mile delivery and cooperative agriculture robot (agribot) fleets.

In this setting, we study the capacitated Vehicle Routing Problem with Profits (VRPP) Aksen and Aras [2006]; Archetti et al. [2014] where a vehicle receives a customer-specific revenue from all visited customers on its route (a dicycle starting from the depot) for which it pays the traveling cost depending on the distance traveled. This paper addresses the scientific question of maximizing the profit of the worst-off vehicle in a fleet entrusted with delivering goods to a set of revenue-generating customers. The solution approach must navigate through multiple constraints, encompassing the vehicles’ restricted autonomy and capacity, as well as the customers’ heterogeneous demands and stringent time windows. Thus, the objective is to develop an approach that optimizes the least profitable vehicle’s route while effectively managing these complex considerations.

With this aim, in this paper, we propose the Vehicle Routing Problem with Fair Profits and Time Windows (VRP-FPTW), which is computationally complex and challenging since it entails efficiently finding routes that consider customers’ hard time windows, and ensure equitable profit distribution among vehicles. For successful implementation in a real-world context, this problem necessitates a robust, distributed, highly scalable, and computationally efficient solution methodology capable of effectively handling large-scale data sets and stringent time constraints. We propose such a solution approach, which is based on problem decomposition through column generation and we implement it in a multi-agent system composed of a fleet coordinator and vehicle agents. This approach

allows for a distributed, parallel, and asynchronous computation, particularly useful for large-scale real-world crowdsourced delivery fleet routing with no need to share private vehicle data. Furthermore, the column generation method guarantees an optimal solution when combined with a Branch and Price method Pecin et al. [2017].

Contrary to the proposed problem, related state-of-the-art VRPP models do not oblige to visit each customer (e.g., Archetti et al. [2014]), and the problem lies in choosing the customers and distributing them among the vehicles in the fleet. Furthermore, they also do not guarantee the fair treatment of customers since the routes are found considering only fairness among vehicles.

The paper is structured as follows. In Section 2, we give an overview of the state of the art. Section 3 presents the (centralized) problem formulation. In Section 4, we decompose the problem by column generation and propose the multi-agent architecture in which the scalable solution approach is implemented. Section 5 describes the simulation setup and presents experiment results with related discussion. We close the paper in Section 6 with conclusions and future work.

2 Preliminaries and related work

Achieving maximum overall benefit is not the only objective considered in collaborative vehicle routing problems (e.g., Guajardo and Rönnqvist [2016], Gansterer and Hartl [2018], Gansterer and Hartl [2020]). Here, vehicles collaborate to increase their profits while ensuring fair cost distribution. In Soriano et al. [2023], Soriano et al. investigate how to solve the Multi-Depot Vehicle Routing Problem (MDVRP) with the addition of profit fairness constraints. They consider each depot as an independent partner with its fleet and propose a bi-objective optimization problem that minimizes the total cost and maximizes the fairness of profit distribution. A multi-objective optimization approach with a max-min objective function that maximizes the owners' lowest satisfaction level was proposed in Liu and Papageorgiou [2018]. Furthermore, Lee and Ahn [2017] introduces a novel vehicle routing problem with vector profits, where each customer's revenue is linked to multiple stakeholders involved. The objective is to maximize the total profit

while satisfying the stakeholder with the lowest level of satisfaction. The authors adopt a column-generation approach to solve the problem.

Column generation is an optimization technique used to solve large-scale linear and mixed-integer programming problems (e.g., Nemhauser [2012]). It involves solving a restricted version of the problem using a subset of the variables and generating new variables (columns) to improve the obtained solution. These variables are generated by identifying a set of dual variables used to improve the solution, and adding them to the problem. This process is repeated until no further improvements can be made. Dantzig–Wolfe decomposition is a technique based on column generation that involves splitting a large optimization problem into smaller subproblems that can be solved independently and in parallel. It works by first formulating the main original constraints of the problem as a master problem and the rest can be formulated as subproblems. These find new columns that may improve the objective function of the master problem.

In addition to considering fairness among vehicles, it is also important to consider fairness among customers. Stavropoulou et al. Stavropoulou et al. [2019] study a VRP with profits and consistency constraints that uses a mixed set of mandatory and optional customers to visit. The objective is to determine profitable vehicle routes that maximize the net profit while satisfying vehicle capacity, route duration, and consistency constraints. Moreover, Mancini et al. Mancini et al. [2021] and Rodríguez-Martín et al. Rodríguez-Martín et al. [2019] discuss how to distribute the customer’s requirements in multiple days with the addition of driver consistency constraints where customers must be served by the same vehicle over time.

The above approaches consider a centralized approach, where all information is shared. Lujak et al. in Lujak et al. [2014] study a distributed solution approach for the traffic assignment problem where routes for individual vehicles are found considering congestion and fairness in the cost of the routes, while in Lujak and Giordani [2018], they propose a distributed algorithm for route finding considering fairness in the distribution of the costs of the routes in evacuation and maximizing the robustness of the routes, in terms of the number of routes with similar costs.

The present work is a continuation of our previous work (López Sánchez et al. [2022, 2023]), where we proposed a model for balancing the fairness and efficiency of vehicles' routes when no customer constraints apply. This paper proposes a more sophisticated model that introduces customers' time windows (to specify when a certain customer has to be visited) and resource requirements. Furthermore, we propose a decomposition approach based on column generation, which allows for distributing the decision-making process in a multi-agent system context.

3 Problem Formulation

Given is a set of customers to be visited, $N = \{1, \dots, |N|\}$. To represent the transportation network, we consider a complete arc-weighted digraph $G = (V, A)$, where $V = \{0, \dots, |N|\}$ is a set of vertices formed by combining the depot $\{0\}$ and the customers N . A represents the set of arcs $(i, j) \in V \times V$ with $i \neq j$, indicating the shortest paths between any two distinct vertices i and j in the transportation network graph G . The distance of arc $(i, j) \in A$ is denoted as d_{ij} .

Each customer $i \in N$ is associated with a demand q_i to be satisfied and a nonnegative revenue r_i , while a revenue $r_0 = 0$ is associated with the depot. The revenue of each customer may be collected at most once. A time window $[l_i, u_i]$ is associated with each customer $i \in N$ to specify the interval in which i should be visited. A fleet K of $|K|$ heterogeneous vehicles is initially located in the depot vertex 0. Each vehicle $k \in K$ travels with velocity s_k , with an autonomy D_k (maximum traveled distance between two visits to the depot), a resource capacity Q_k , and a travel cost per unit of distance travelled o_k (i.e., we consider heterogeneous vehicles). The travel time from vertex i to j for vehicle k is $t_{ijk} = d_{ij}/s_k$. Furthermore, we denote the cost for vehicle k to travel from vertex i to j by $c_{ijk} = d_{ij} * o_k$.

The mission of the fleet is to collaboratively visit all customers once, each one in the specified time interval, and satisfy its specified demand while leaving from and returning to the single depot. Each vehicle has attributed a profit, which is derived by computing the difference between the total revenue generated from the customers it serves and the

Table 1: VRP-FPTW notation

Parameter	Symbol	Description
<i>Fleet's shared information</i>		
Customers	N	Set of customer vertices to visit, $N = \{1, \dots, N \}$.
Depot	0	Depot.
Vertices	V	Set of vertices $i, j \in \{0\} \cup N$.
Arcs	A	Set of arcs $(i, j) \in V \times V$, where $i \neq j$.
Distances	d_{ij}	Travel distance of arc $(i, j) \in A$.
Revenue	r_i	Nonnegative revenue obtained by visiting customer $i \in N$.
Demand	q_i	Nonnegative demand associated with each customer $i \in N$.
Time window	$[l_i, u_i]$	Time window to visit customer i .
Vehicles	K	Set of vehicles $K = \{1, \dots, K \}$, composed of $ K $ vehicles.
<i>Vehicle's private information</i>		
Vehicle cost	c_{ijk}	Cost related with vehicle k to travel edge $(i, j) \in A$.
Travel time	t_{ijk}	Travel time spent by vehicle k on edge $(i, j) \in A$.
Autonomy	D_k	Autonomy of vehicle k in terms of maximum traveled distance.
Capacity	Q_k	Capacity of vehicle k .
Decision variable	Symbol	Description
Route variable	x_{ijk}	$x_{ijk} = 1$ if vehicle $k \in K$ travels from vertex $i \in V$ to vertex $j \in V$ and 0 otherwise.
Time variable	v_i	Expresses the time in which customer i is visited

overall cost of its route (directed cycle originating and concluding at the depot). In this context, the objective of the VRP with fair profits and time windows (VRP-FPTW), which we propose next, is to find a set of routes, one for each vehicle, that maximize the profit of the worst-off vehicle while the constraints on customers' demands and time windows, as well as the vehicles' capacities and autonomies, are satisfied. Table 1 resumes the parameters and decision variables of the proposed (Centralized) Mixed Integer Linear Programming model:

$$\max \quad y \tag{1}$$

$$s.t. \quad \sum_{k \in K} \sum_{i \in V} x_{ijk} = 1 \quad \forall j \in N, \tag{2}$$

$$\sum_{i \in V} x_{ijk} - \sum_{h \in V} x_{jhk} = 0 \quad \forall j \in V, \forall k \in K, \tag{3}$$

$$\sum_{j \in N} x_{0jk} \leq 1 \quad \forall k \in K, \tag{4}$$

$$\sum_{i \in V} \sum_{j \in V} d_{ij} x_{ijk} \leq D_k \quad \forall k \in K, \tag{5}$$

$$\sum_{i \in V} \sum_{j \in N} q_j x_{ijk} \leq Q_k \quad \forall k \in K, \tag{6}$$

$$v_i + t_{ijk} \leq v_j + M(1 - x_{ijk}) \quad \forall (i, j) \in A, j \neq 0, \tag{7}$$

$$\forall k \in K,$$

$$l_i \leq v_i \leq u_i \quad \forall i \in N, \tag{8}$$

$$y \leq \sum_{i \in V} \sum_{j \in V} (r_i - c_{ijk}) x_{ijk} \quad \forall k \in K, \tag{9}$$

$$y \in \mathbb{R}, x_{ijk} \in \{0, 1\} \quad \forall (i, j) \in A, \forall k \in K, \tag{10}$$

where x_{ijk} are binary decision variables equal to 1 if arc (i, j) is used by vehicle k , and 0, otherwise. Continuous variable v_i indicates the time of arrival of a vehicle at customer vertex $i \in V$. The objective function (1) combined with constraints (9) ensure that the profit of the worst-off vehicle, here assigned to variable y , is maximized. Constraints (2) ensure that each customer is visited once by exactly one vehicle. Flow conservation constraints (3) state that if a vehicle visits a vertex, it should also leave it. Constraints (4) ensure that each vehicle is used at most once. Constraints (5) and (6) ensure that

the total travel time and resource demands of the customers to be visited by any vehicle $k \in K$ do not exceed its autonomy D_k nor capacity Q_k , respectively. We consider vehicles that dispose of an electric motor, internal combustion engine, etc., the charging of which is not immediate. Therefore, the length of the longest directed cycle (dicycle) traveled by vehicle k has to be equal to or lower than the autonomy of that vehicle in terms of the maximum traveled distance. Finally, constraints (7) assure that vehicle routes start and finish at the depot while (8) ensure that the time windows are satisfied. The value of M represents the maximum allowable time sufficient for returning to the depot from any customer location.

The proposed VRP-FPTW model is a highly versatile formulation that can accommodate both heterogeneous and homogeneous vehicles. It imposes no restrictions on the graph's structure, the number of vehicles, or the number of customers. However, it is crucial to ensure that a sufficient number of vehicles with their related autonomy and capacity values are available to meet the customers' constraints. Otherwise, a feasible solution might not exist for a given problem. In such cases, the application of data pre-processing techniques becomes necessary to detect and effectively manage these situations.

4 Distributed MAS solution approach

Our focus lies in developing a distributed multiagent system (MAS) algorithm to solve the proposed VRP-FPTW problem. To accomplish this, we propose and implement a solution approach within a multi-agent architecture consisting of two types of agents: a fleet coordinator agent and multiple vehicle agents (one agent per vehicle), as depicted in Figure 1. The proposed solution approach is based on the column generation technique, where the fleet coordinator agent adopts the restricted master problem, while each vehicle agent adopts its subproblem based on its private information including its own costs and distances of the graph, autonomy, capacity, speed, and travel time. The communication between the coordinator and vehicles can be seen as a coordination process, where the coordinator agent iteratively sends information about the vehicles' routes (shadow prices)

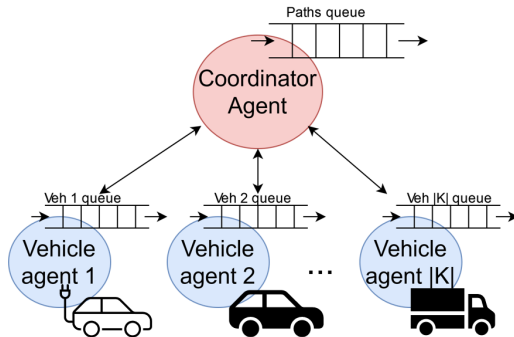


Figure 1: Multi-agent architecture.

to vehicle agents and the vehicle agents generate new feasible routes that take into account the shadow prices and then send them back to the coordinator. At the end of the process, the coordinator is in charge of assigning a single route for each vehicle. The route selection is done considering the best routes communicated by every vehicle. A description of each agent is given next.

4.1 Fleet coordinator agent

The problem that the fleet coordinator agent solves is the restricted master problem, which is achieved by using a Dantzig-Wolfe decomposition for integer programs Lübbecke and Desrosiers [2005]; Pinto et al. [2015] of the centralized problem (1)-(10). This leads to a route-based formulation, where a route is a directed cycle. This is because constraints (3)-(8) are formulated independently and define the properties that a route must follow for each vehicle $k \in K$ and constraints (2) and (9) gather the route properties of all vehicles. Therefore, each vehicle $k \in K$ finds a set of the best feasible routes $p \in \Omega_k$ and communicates them to the coordinator, if necessary. The coordinator agent receives from each vehicle its best routes that can be seen as columns of the restricted master problem (11) –(15) and assigns one of these routes to each vehicle such that the profit of the worst-off vehicle is maximized. Thus, considering the elements of the decomposed problem in Table 2, the coordinator agent must solve the following problem, determining the values of binary decision variables λ_k^p .

$$\max \quad y \tag{11}$$

$$s.t. \quad \sum_{k \in K} \sum_{p \in \Omega_k} a_{jk}^p \lambda_k^p = 1, \forall j \in N, \tag{12}$$

$$\sum_{p \in \Omega_k} w_k^p \lambda_k^p \geq y, \quad \forall k \in K, \tag{13}$$

$$\sum_{p \in \Omega_k} \lambda_k^p = 1, \quad \forall k \in K, \tag{14}$$

$$y \in \mathbb{R}, \lambda_k^p \in \{0, 1\}, \forall k \in K, \forall p \in \Omega_k. \tag{15}$$

where constraints (12) are the one-on-one vehicle-customer assignment constraints; fairness constraints (13) fix the minimum profit for each vehicle to y ; vehicle constraints (14) guarantee the use of all vehicles, while the objective function (11) maximizes profit y of the worst-off vehicle.

Table 2: Decomposed VRP-FPTW notation

Set	Symbol	Description
Customers	N	Set of customers $\{1, \dots, V - 1\}$.
Vehicles	K	Set of vehicles $K = \{1, \dots, K \}$, where there are $ K $ vehicles available
Vehicle routes	Ω_k	Set of feasible routes for vehicle k .
Parameter	Symbol	Description
Customer indicator	a_{jk}^p	$a_{jk}^p = 1$ indicates if the customer $j \in N$ is visited on route p by vehicle k , and 0 otherwise.
Route profit	w_k^p	Profit obtained on route p by vehicle k .
Variable	Symbol	Description
Route variable	λ_k^p	$\lambda_k^p = 1$ if route p is selected for vehicle k and 0 otherwise.

Mathematical model (11)–(15) is used by the coordinator agent both for vehicle-route assignment and to calculate the shadow prices (dual variables) of a given assignment. The shadow prices in this context are obtained by solving the linear relaxation of the restricted master problem and its dual Zilinskas [2007]. They represent the marginal value or cost associated with relaxing or modifying a constraint. They provide an economical

interpretation in terms of the opportunity cost of resources and can be both positive and negative.

A positive shadow price means that the objective function will increase by increasing the right-hand side (RHS) of the constraint by one unit. It implies that the constraint is binding, and allocating additional resources to it will have a beneficial impact on the overall objective. On the other hand, a negative shadow price suggests that the objective function will decrease by increasing the RHS by one unit. It indicates that the constraint is not binding, and allocating more resources to it would result in a suboptimal outcome. Specifically, the assignment constraints (12) produce marginal costs $\pi_j \in \mathbb{R}$ that indicate the marginal cost of attending customer $j \in N$ in a route. Similarly, the fairness constraints (13) and vehicle constraints (14) produce $\mu_k \leq 0$ and $\alpha_k \in \mathbb{R}$ that represent the marginal cost for a particular vehicle $k \in K$.

After receiving the routes from the vehicles, the coordinator agent calculates their shadow prices and sends them back this information. Each vehicle agent receives shadow prices and uses them to possibly find new route(s) that it sends to the coordinator to add them to feasible routes for that vehicle. This process is repeated iteratively until no vehicle can find new, better route(s). Then, the coordinator agent solves the restricted master problem and assigns final routes to the vehicle agents. The process is presented in Algorithm ???. Here, only the marginal costs of customers π_j are shared information among all agents. Marginal costs μ_k and α_k of a particular agent k are only sent by the coordinator to vehicle agent k . Each vehicle only sends the set of its best feasible routes $p \in \Omega_k$ together with their related profits w_k^p . Thus, privacy is preserved.

4.2 Vehicle agent

Each vehicle $k \in K$ calculates new "interesting" routes (new columns, which have the potential to improve the current value of the objective function) incited by new shadow prices received from the coordinator agent. The "columns" refer to decision variables in the problem formulation. Each column represents a potential solution or configuration of values for the variables.

Algorithm 1 Coordinator agent algorithm

- 1: Set initial routes
 - 2: **repeat**
 - 3: Solve the linear relaxation of the restricted master problem and calculate shadow prices ($\forall j \in N : \pi_j, \forall k \in K : \mu_k$ and α_k)
 - 4: For all $k \in K$: Send π_j, μ_k and α_k to vehicle k
 - 5: Wait for new route proposal ($\langle p, w_k^p \rangle$) from any agent k
 - 6: $\Omega_k = \Omega_k \cup \{p\}$
 - 7: **until** No more route proposals;
 - 8: Solve the Restricted Master Problem
 - 9: For all $k \in K$: Send p to k such that $\lambda_k^p = 1$
-

Each vehicle agent solves its subproblem by calculating its new "interesting" routes with a negative reduced cost. When vehicle k receives shadow prices π_j, μ_k and α_k from the coordinator, it updates the costs of its graph arcs as follows:

- $\hat{c}_{ijk} = (\pi_j + (r_i - c_{ijk})\mu_k), \forall (i, j) \in V \times N$, the new reduced cost for arcs $(i, j) \in V \times N$.
- $\hat{c}_{i0k} = (r_i - c_{i0k})\mu_k + \alpha_k, \forall i \in V$, the new reduced cost for the arcs to return to the depot.

The mathematical formulation of each vehicle subproblem is defined by constraints (1)-(10). They correspond to the elementary shortest path problem with resource constraints (ESPPRC) Dror [1994], which finds routes with the minimum reduced cost ensuring that the resource constraints and time windows are respected. Because of the significant computational complexity associated with the ESPPRC problem, we have implemented sequentially two distinct algorithms for vehicles. In each iteration, vehicles send to the coordinator all negative cost routes they find. The first algorithm is a labeling heuristic based on the SPPRC, which prohibits cycles and uses a simple dominance strategy, where a route is better than another if its accumulated weight is lower Irnich and Villeneuve [2006]. In the last iteration, to demonstrate that there are no additional routes with reduced cost, the second algorithm implements an exact approach based on Feillet et al. [2004]. In summary, each vehicle optimizes its own subproblem in a distributed manner, focusing on the reduced costs in its graph, while the coordinator agent

add this routes asynchronously and coordinates the overall assignment of customers and maintains fairness between vehicles.

The communication between the coordinator and vehicle agents can be accomplished in an asynchronous way, that is, the coordinator can calculate new shadow prices whenever it receives any new route proposal. This can easily be implemented through message queues and has the additional advantage that the system will become more robust with regard to possible breakdowns in the communication channel or other contingencies. In general, the proposed approach to distribute the decision-making process through a multi-agent architecture will have the following benefits: 1. **Privacy**. Vehicles are independent agents that make decisions based on local information and their private data is inaccessible to others. 2. **Effectiveness**. The approach facilitates effective decision-making by enabling each vehicle to optimize its route subject to its individual constraints. 3. **Robustness**, by enabling incorporation or elimination of vehicles at runtime, even when some vehicles or customers fail or become unavailable.

5 Functional experiments

In this Section, we compare the performance of the proposed distributed MAS approach considering a coordinator and vehicle agents (MAS) with the centralized model (Centr.). The simulation experiments were run on an Intel Xeon Gold 6226R* virtualized cluster with 16 CPUs, 32 GB of RAM and clock frequency 2.9 GHz. The whole process was run in the Gurobi solver with gurobipy Python library.

Experiment setup In our experiments, the set of customers is uniformly distributed in a 2D square environment $[0, 30]^2$ with the deposit positioned centrally (15,15). The demand of each customer node is distributed uniformly randomly between 1 and 10, as well as their time windows, ranging from 10 to 50. The maximum time to visit all the customers is found as the time to travel around the perimeter of the square plus the time

* <https://www.intel.es/content/www/es/es/products/sku/199347/intel-xeon-gold-6226r-processor-22m-cache-2-90-ghz/specifications.html>

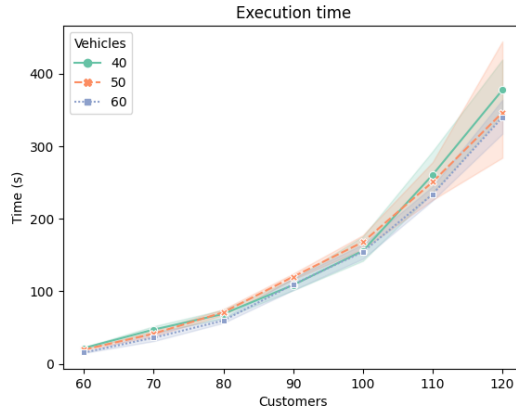


Figure 2: Times obtained with the distributed MAS algorithm

to travel from the deposit to one side and return to the deposit from the other side of the square. The sum of the revenue of the customers is fixed per each instance to $(|V| - 1)^2$. For each graph structure, we create 10 instances, distributing this total revenue among the customers randomly. The parameters' values are chosen to limit vehicle routes in autonomy and capacity, considering customers' demand and time windows.

Execution time analysis To show the execution time of our distributed MAS algorithm we test it in instances with 60 to 120 customers, with 40, 50, and 60 vehicles. In Figure 2 we show the evolution time of the algorithm, changing the number of customers. The presented results are the mean of the 10 different instances with every combination of customers and vehicles. The Figure shows that the execution time increases non-linearly with the number of customers, but not with the number of vehicles. Generally, the centralized (Centr.) model is not able to get the optimal solution within an hour, so its execution time is always the user time limit.

Comparison of the centralized model and the MAS algorithm with sort times

In Table 3 we show the results obtained with the two models in a 5 min (300 s) execution limit, simulated fast-reaction capability in real-life environments. For each instance and method, we indicate the worst vehicle's profit, which is the objective value of the max-min problem achieved in one hour (y), the computation time in seconds (Time), and

the number of instances that have been solved (not necessarily to optimality) within one hour (Count) where X/Y is read X out of Y. We notice first that the MAS algorithm always gets a solution for all instances, but the centralized algorithm (Centr.) starts failing with 80 customers and 40 vehicles. With 100 customers and 40 vehicles, the centralized model is not able to find a solution for any instance in less than 5 min. The income column shows that in instances with a lower number of customers (60 and 70), the centralized model achieves better results in the objective function value, but when the number of customers increases, the MAS algorithm is significantly better. When the number of customers is higher, the MAS algorithm always achieves a positive profit for the worst-off vehicle, but the centralized model underperforms.

Lastly, in all the experiments, the MAS algorithm achieves a better best bound (BBound)[†]. When this value is equal to the income y , the solution is optimal. This value is used to compare how good a solution with optimality guarantees is when the target value is negative or zero, as the gap is not defined. To achieve a better solution, the MAS algorithm relying on column generation can be introduced into a branch-and-price algorithm to find the optimal solution.

Time-dependent results If we allow more time to run the centralized model with 100 customers, it will return an improved solution or a better best bound due to its implementation as a MIP model with Gurobi. At the end of Table 3, we show the results of the centralized model after 300, 600, and 3600 seconds with 100 customers, and compare them with the results obtained with the MAS algorithm. The MAS algorithm can always obtain a solution in the same instances as the centralized model since the latter is computationally expensive and is not able to find feasible solutions quickly. The income column shows that with 10 minutes time limit, the centralized model achieves worse solutions for the worst-off vehicle compared with the MAS algorithm. When the time limit is one hour, the centralized model gives a better solution with 40 and 50 vehicles but when the number of vehicles increases to 60, the MAS algorithm achieves

[†]<https://www.gurobi.com/documentation/9.1/refman/mipgap.html>

Table 3: Results.

#Cust.	#Veh.	Algorithm	y	BBound	Time	Count
60	40	MAS	10.87	72.00	21.74	10/10
		Centr.	2.02	80.62	300	10/10
	50	MAS	-8.06	54.54	19.52	10/10
		Centr.	-2.88	64.07	300	10/10
	60	MAS	-18.24	42.88	15.47	10/10
		Centr.	-10.74	50.66	300	10/10
70	40	MAS	28.54	104.22	47.18	10/10
		Centr.	30.73	111.97	300	10/10
	50	MAS	10.39	80.03	41.52	10/10
		Centr.	4.79	89.19	300	10/10
	60	MAS	-10.98	63.66	36.02	10/10
		Centr.	0.00	72.35	300	10/10
80	40	MAS	56.55	141.31	68.76	10/10
		Centr.	18.27	149.18	300	5/10
	50	MAS	29.63	109.92	71.02	10/10
		Centr.	0.00	119.40	300	10/10
	60	MAS	-0.81	88.88	59.62	10/10
		Centr.	-8.14	98.60	300	10/10
90	40	MAS	70.99	183.18	108.58	10/10
		Centr.	50.58	190.56	300	2/10
	50	MAS	52.79	143.94	120.00	10/10
		Centr.	-1.47	152.45	300	8/10
	60	MAS	38.66	116.94	109.37	10/10
		Centr.	-15.83	126.54	300	10/10
100	40	MAS	103.76	231.77	147.22	10/10
		Centr.	—	237.25	300	0/10
			83.60	237.25	600	7/10
	50		164.96	237.25	3600	7/10
		MAS	75.38	182.02	167.83	10/10
		Centr.	-27.46	189.84	300	1/10
	60		-5.26	189.73	600	10/10
			106.02	189.72	3600	10/10
		MAS	54.18	149.04	154.27	10/10
		Centr.	-16.74	157.89	300	10/10
			-12.52	157.82	600	10/10
			49.64	157.79	3600	10/10

better results. Lastly, the same as before, in all the experiments, the MAS algorithm achieves a better best bound.

6 Conclusions and future work

In this work, we proposed the Vehicle Routing Problem with Fair Profits and Time Windows (VRP-FPTW) designed for application in crowdsourced delivery organizations. The objective of the VRP-FPTW is to maximize the profit of the worst-off vehicle in the fleet while visiting all customers in their required time windows and while respecting their demands and vehicles' autonomy and capacity. We addressed this problem by proposing a centralized and a decomposed VRP-FPTW mathematical model.

The proposed decomposed model involves breaking down the centralized VRP-FPTW model into smaller components implemented in a distributed multi-agent system composed of a fleet coordinator agent and multiple vehicle agents. The objective of the coordinator agent is to assure that the fleet's norms are respected and that all the vehicles are assigned a route and all the customers are served while maximizing the worst-off vehicle's profit. The objective of the individual vehicles is to find the best routes based on their private objectives and constraints and the shadow prices received from the coordinator agent.

The vehicle agents perform iterative computations to determine individually beneficial routes based on the momentary shadow prices received from the coordinator agent. The shadow prices provide information about the costs and benefits associated with specific routes. Once the vehicle agents compute their routes, they share them with the coordinator agent who is responsible for considering the requirements and norms of the fleet as a whole. It iteratively evaluates the routes provided by the vehicle agents and assigns them to the vehicles once it ensures that they adhere to the fleet requirements.

Compared to the centralized model, the distributed model provides several advantages. Firstly, it offers greater robustness, meaning it is more resilient and can handle disruptions more effectively. This robustness stems from its decentralized nature, as the computation is distributed among the vehicle agents and the fleet coordinator agent,

reducing the dependency on a single central entity. Secondly, the distributed model improves fairness by ensuring a more equitable distribution of profit among vehicle agents. This fairness is further achieved by allowing each vehicle agent to contribute to the fleet’s decision-making process, enabling a more inclusive and democratic approach without sharing the vehicle’s private information. Additionally, the distributed model reduces computation time. By leveraging the parallel processing capabilities of multiple agents, the computational workload is divided, enabling concurrent computation and faster results, which may be crucial in dynamic environments.

Furthermore, the distributed model excels in finding better suboptimal routing solutions within a reasonable timeframe. While it may not guarantee an optimal solution, it is designed to quickly generate high-quality routing solutions with quality-of-solution guarantees. This is valuable in situations where finding the optimal solution may be computationally infeasible or time-consuming, allowing for efficient routing decisions.

The proposed decomposed VRP-FPTW model and related solution approach are of concern for real-world applications where each vehicle disposes of private information that it does not intend to share with others and the decision-making time for the generation of routes is short. The proposed MAS-based distributed solution approach is scalable and is suited for large fleets. However, it assumes non-strategic vehicle agents that share truthful information about their routes, which might not be the case with strategic agents that may deceive information to obtain higher profits.

In our future work, we plan to explore strategy-proof methods that account for strategic agents in the context of our study. One potential approach is the use of strategy-proof mechanism design methods, such as Vickrey auctions. These methods have been widely studied in economics and can provide insights into designing mechanisms that encourage participants to reveal their true preferences and act honestly.

Additionally, we intend to introduce the receding horizon control into the proposed approach, so that the most disadvantaged vehicles in one period are prioritized in the following one. The dynamic prioritization of disadvantaged vehicles can encourage cooperation and participation in the fleet.

To further improve the fleet’s efficiency, we plan to engineer the proposed method with branch and price methods to find close-to-optimal solutions in a reasonable amount of time. Finally, we will explore the application of systematic egalitarian algorithms, as presented in López Sánchez et al. [2023], to improve the overall efficiency of the system.

Acknowledgments

This work was supported by grant VAE: TED2021-131295B-C33 funded by MCIN/AEI/ 10.13039/501100011033 and by the “European Union NextGenerationEU/PRTR”, by grant COSASS: PID2021-123673OB-C32 funded by MCIN/AEI/ 10.13039/501100011033 and by “ERDF A way of making Europe”, and by the AGROBOTS Project of the University Rey Juan Carlos funded by the Community of Madrid, Spain.

References

- Aksen, D. and Aras, N. (2006). Customer selection and profit maximization in vehicle routing problems. In *Operations research proceedings 2005*, pages 37–42. Springer.
- Archetti, C., Speranza, M. G., and Vigo, D. (2014). Chapter 10: Vehicle routing problems with profits. In *Vehicle routing: Problems, methods, and applications, second edition*, pages 273–297. SIAM.
- Dror, M. (1994). Note on the complexity of the shortest path models for column generation in vrptw. *Op. Res.*, 42(5):977–978.
- Feillet, D., Dejax, P., Gendreau, M., and Gueguen, C. (2004). An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. *Networks: An International Journal*, 44(3):216–229.
- Gansterer, M. and Hartl, R. F. (2018). Collaborative vehicle routing: a survey. *European J. of Op. Res.*, 268(1):1–12.

- Gansterer, M. and Hartl, R. F. (2020). Shared resources in collaborative vehicle routing. *Top*, 28:1–20.
- Guajardo, M. and Rönnqvist, M. (2016). A review on cost allocation methods in collaborative transportation. *International transactions in operational research*, 23(3):371–392.
- Irnich, S. and Villeneuve, D. (2006). The shortest-path problem with resource constraints and k-cycle elimination for $k \geq 3$. *INFORMS Journal on Computing*, 18(3):391–406.
- Lee, D. and Ahn, J. (2017). Vehicle routing problem with vector profits (vrpvp) with max-min criterion. *arXiv preprint arXiv:1710.10550*.
- Liu, S. and Papageorgiou, L. G. (2018). Fair profit distribution in multi-echelon supply chains via transfer prices. *Omega*, 80:77–94.
- López Sánchez, A., Lujak, M., Semet, F., and Billhardt, H. (2022). On balancing fairness and efficiency in routing of cooperative vehicle fleets. In *CEUR Proc., 12th Int. Workshop ATT@IJCAI-ECAI 2022*, volume 3173.
- López Sánchez, A., Lujak, M., Semet, F., and Billhardt, H. (2023). How to achieve fair and efficient cooperative vehicle routing? models and algorithms. *AI Comm., IOS-Press*, Submitted.
- Lübbecke, M. E. and Desrosiers, J. (2005). Selected topics in column generation. *Op. Res.*, 53(6):1007–1023.
- Lujak, M. and Giordani, S. (2018). Centrality measures for evacuation: Finding agile evacuation routes. *Future Generation Computer Systems*, 83:401–412.
- Lujak, M., Giordani, S., and Ossowski, S. (2014). Fair route guidance: Bridging system and user optimization. In *17th International IEEE Conference on ITSC*, pages 1415–1422. IEEE.

- Mancini, S., Gansterer, M., and Hartl, R. F. (2021). The collaborative consistent vehicle routing problem with workload balance. *European J. of Op. Res.*, 293(3):955–965.
- Nemhauser, G. L. (2012). Column generation for linear and integer programming. *Optimization Stories*, 20(65):U73.
- Pecin, D., Pessoa, A., Poggi, M., and Uchoa, E. (2017). Improved branch-cut-and-price for capacitated vehicle routing. *Mathematical Programming Computation*, 9:61–100.
- Pinto, T., Alves, C., Mansi, R., and Valério de Carvalho, J. (2015). Solving the multi-scenario max-min knapsack problem exactly with column generation and branch-and-bound. *Mathematical Problems in Engineering*, 2015.
- Rodríguez-Martín, I., Salazar-González, J.-J., and Yaman, H. (2019). The periodic vehicle routing problem with driver consistency. *European J. of Op. Res.*, 273(2):575–584.
- Soriano, A., Gansterer, M., and Hartl, R. F. (2023). The multi-depot vehicle routing problem with profit fairness. *International Journal of Production Economics*, 255:108669.
- Stavropoulou, F., Repoussis, P. P., and Tarantilis, C. D. (2019). The vehicle routing problem with profits and consistency constraints. *European J. of Op. Res.*, 274(1):340–356.
- Zilinskas, A. (2007). Conejo, antonio, enrique castillo, roberto minguez, raquel garcia-bertrand. decomposition techniques in mathematical programming: Engineering and service applications. *Interfaces*, 37(5):488–491.