Luiz Felipe de Oliveira Moura Santos

Practical Pollution-Routing Problem with time-dependent speeds: insights from São Paulo

São Paulo

2019

Luiz Felipe de Oliveira Moura Santos

Problema Prático de *Pollution-Routing* com velocidades dependentes do horário: percepções de São Paulo

São Paulo

2019

Luiz Felipe de Oliveira Moura Santos

Practical Pollution-Routing Problem with time-dependent speeds: insights from São Paulo

Dissertation presented to the Polytechnic School of the University of São Paulo to obtain the Master of Science degree in Production Engineering.

Research area:
Operations and Logistics Management

Advisor:
Prof. Hugo T. Y. Yoshizaki, PhD

São Paulo
2019

Catalogação-na-publicação

Santos, Luiz Felipe de Oliveira Moura

Practical Pollution-Routing Problem with time-dependent speeds: insights from São Paulo / L. F. O. M. Santos – São Paulo, 2019.
100 p.

Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Produção.

1.Roteirização 2.Poluição 3.Transporte Urbano 4.Otimização Combinatória 5.Heurística I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Produção II.t.

Luiz Felipe de Oliveira Moura Santos

Problema Prático de *Pollution-Routing* com velocidades dependentes do horário: percepções de São Paulo

Dissertação apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do título de Mestre em Ciências em Engenharia de Produção.

Área de concentração:
Gestão de Operações e Logística (GOL)

Orientador:
Prof. Dr. Hugo Tsugunobu Yoshida Yoshizaki

São Paulo
2019

Catalogação-na-publicação

Santos, Luiz Felipe de Oliveira Moura

Practical Pollution-Routing Problem with time-dependent speeds: insights from São Paulo / L. F. O. M. Santos – São Paulo, 2019.
100 p.

Dissertação (Mestrado) – Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Produção.

1.Roteirização 2.Poluição 3.Transporte Urbano 4.Otimização Combinatória 5.Heurística I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Produção II.t.

Acknowledgments

First of all I am very thankful to my family, specially my mother, father, sister and brother, for all of them being so supportive from the start of this journey. Thanks also to my companion Lucy, who helped me several times, and encouraged me in difficult moments and those in which I couldn't be very present.

I am also thankful to my advisor Prof. Hugo Yoshizaki, for guiding this research and for all of his dedication, trust and encouragement throughout the development of this research. I also acknowledge everyone at CISLog, for the relevant discussions and friendships along these years together.

Thanks to all my professors and lecturers from the Polytechnic School of the University of São Paulo, who shared valuable knowledge that led to the accomplishment of this research, in special to Prof. Claudio B. Cunha, for all of the opportunities, rich discussions and relevant insights.

I also acknowledge the use of computational facilities of LCCA-Laboratory of Advanced Scientific Computation of the University of São Paulo (LCCA-USP) and the Brazilian National Council for Scientific and Technological Development (CNPq) financial support [grant number 830616/1999-3].

Abstract

Vehicle routing problems (VRP) are one of the most studied problems in combinatorial optimization. In recent years, it has gained the attention of many researchers and business organizations not only in its classical application, but also because of the many environmental considerations that it is possible to consider. Objectives that aims to reduce fuel consumption or directly the emissions of CO_2 and other greenhouse gases (GHG) have become widely adopted. Being so, many models that estimate emissions and fuel consumption have been proposed in the literature, accounting for different factors that affect fuel consumption in road transportation, such as payload, slope and travel speed. These comprehensive models, while providing very accurate results in modeling emissions and fuel consumption, are very dependent on numerous parameters and user inputs, many not easily retrievable by practitioners. In this way, a Practical Pollution-Routing Problem (PPRP) is a VRP that aims to reduce fuel consumption using simple and practical computations based on the Fuel Consumption Rate (FCR) of vehicles. Another important variant of the VRP that has also attracted recent attention is the time-dependent vehicle routing problem (TDVRP), in which the effects of congestion and travel speeds fluctuations during the day are taken into account when designing the delivery routes. A time-dependent approach to the PPRP, however, has not been found in the literature. Thus, our research addresses this gap and aims to propose a new variant of this problem called the Practical Pollution-Routing Problem with Time-Dependent speeds (PPRP-TD), as it has important value to practitioners interested in the reduction of fuel consumption and the hazardous effects caused by diesel trucks. We propose several instance problems based on real operations of a major retailer company that distributes in São Paulo, and also analyze the time dependent speeds of the São Paulo network. To solve both the PPRP and the PPRP-TD, we propose the GRASP-FESA solution method, which is an FCR-based Extended Savings Algorithm (FESA) combined with a Greedy Randomized Adaptive Search Procedure (GRASP) metaheuristic. In the case of the PPRP-TD, we also perform an extensive time-dependent scheduling for all given routes. The method is able to provide good results, although being computationally expensive in the case of large instances.

Keywords: Routing. Pollution. Urban transport. Combinatorial optimization. Heuristic.

Resumo

Problemas de roteirização de veículos (VRP) são alguns dos mais estudados problemas na otimização combinatória. Recentemente, pesquisadores e organizações também tem prestado atenção não apenas em suas aplicações clássicas, mas também nas muitas considerações ambientais que podem ser consideradas. Objetivos que buscam reduzir o consumo de combustíveis fósseis e a emissão de gases do efeito estufa tem sido amplamente utilizados. Assim sendo, diversos modelos para estimar as emissões de gases poluentes e consumo de combustível vem sendo propostos, considerado diversos fatores que influenciam no consumo de combustível, como a carga transportada, inclinação da via e velocidade de tráfego. Esses modelos compreensivos, embora forneçam resultados bem precisos na modelagem de emissões e consumo de combustível, são muito dependentes de uma grande quantidade de parâmetros e entradas dos usuários, sendo que muitos não são facilmente obtidos. Dessa forma, um Problema de *Pollution-Routing* Prático é um VRP que busca reduzir o consumo de combustível usando cálculos simplificados para o consumo de combustível, baseados na taxa de consumo de combustível de diferentes classes de veículos. Outra importante variante do VRP é o VRP dependente do horário, em que os efeitos de congestionamento e velocidades de tráfego flutuantes ao longo do dia são considerados ao se construir as rotas de entrega. Entretanto, uma abordagem ao PPRP com velocidades dependentes do horário não foi encontrada na literatura. Portanto, nossa pesquisa endereça esta lacuna e busca propor uma nova variante deste problema, chamado Problema de *Pollution-Routing* Prático com velocidades Dependentes do Horário (PPRP-TD), devido ao importante valor para empresas e praticantes interessados em reduzir o consumo de combustível de suas frotas, assim como os graves efeitos causados ao ambiente por veículos movidos a combustíveis fósseis. São propostas diversas instâncias baseadas nas operações reais de um grande varejista em São Paulo, além de uma análise das velocidades deptendentes do horário na rede de São Paulo. Para resolver tanto o PPRP quanto o PPRP-TD, é proposto o método de solução GRASP-FESA, que é um FCR-based Extended Savings Algorithm (FESA) combinado com a metaheurística GRASP (Greedy Randomized Adaptive Search procedure). No caso do PPRP-TD, também é realizada uma programação extensiva dependente do horário para todas as rotas. O método é capaz de obter bons resultados, embora seja computacionalmente caro no caso de grandes instâncias.

Palavras-chave: Roteirização. Poluição. Transporte urbano. Otimização combinatória. Heurística.

List of Figures

List of Tables

List of Acronyms and Abbreviations

2E-CVRP	Two-Echelon Capacitated Vehicle Routing Problem
AFV	Alternative Fuel-powered Vehicle
ALNS	Adaptive Large Neighborhood Search
BTL-VRPTW	Bi-objective Time, Load and path-dependent Vehicle Routing Problem with Time Windows
BVFB	Bi-objective Vehicle routing problem with Forced Backhauls
CISLog	Center for Innovation in Logistics Systems
CMEM	Comprehensive Modal Emission Model
COPERT	Computer *Programme* to calculate Emissions from Road Transportation
CVRP	Capacitated Vehicle Routing Problem
DP	Dynamic Programming
DSOP	Departure time and Speed Optimization Procedure
ESA	Extended Savings Algorithm
E-TDVRP	Emissions-based Time-Dependent Vehicle Routing Problem
EVRP	Vehicle Routing Problem for Emissions minimization
FCR	Fuel Consumption Rate
FESA	FCR-based Extended Savings Algorithm
FIFO	First-In-First-Out
FMEFCM	Four-Mode Elemental Fuel Consumption Model
FSA	FCR-based Savings Algorithm
GA	Genetic Algorithm
GHG	Greenhouse Gasses
GPS	Global Positioning System
GRASP	Greedy Randomized Adaptive Search Procedure
GRASP-FESA	GRASP FCR-based Extended Savings Algorithm
Green STDCVRP	Green Stochastic Time-Dependent Capacitated Vehicle Routing Problem
Green TDCVRP	Green Time-Dependent Capacitated Vehicle Routing Problem
Green VRPs	Green Vehicle Routing Problems (class of VRPs)
GVRP	Green Vehicle Routing Problem (specific VRP variant)
GVRSP	Green Vehicle Routing and Scheduling Problem
HDT	High-Duty Trucks

Hybrid SA-TS	Hybrid Simulated Annealing with Tabu Search
IRCI	Iterated Route Construction and Improvement
JCR	Journal Citation Reports
LB	Lower Bound
LDT	Light-Duty Trucks (used as a synonym for VUC in our research)
LS	Local Search
MDT	Medium-Duty Trucks
MEET	Methodology for calculating Transportation Emissions and Energy consumption
MILP	Mixed Integer Linear Programming
MTHVRPP	Minimal-carbon-footprint Time-dependent Heterogeneous-fleet Vehicle Routing Problem with alternative Paths
NAEI	National Atmospheric Emissions Inventory
NYC	New York City
OHD	Off-Hour Deliveries
OTDVRP	Open Time Dependent Vehicle Routing Problem
PC	Personal Computer
PPRP	Practical Pollution-Routing Problem
PPRP-TD	Practical Pollution-Routing Problem with Time-Dependent Speeds
PRP	Pollution-Routing Problem
RCL	Restricted Candidate List
R-FESA	Randomized FESA
RMSP	São Paulo Metropolitan Region
RSFCM	Running Speed Fuel Consumption Model
SA	Simulated Annealing
SMSAH	String-Model-based Simulated Annealing with Hybrid exchange rule
SJR	SCImago Journal Rank
SOP	Speed Optimization Procedure
SP	São Paulo
SPZO	*São Paulo Zona Oeste* (São Paulo's West Zone)
TD	Time-Dependent
TD Scheduling	Time-Dependent Scheduling
TDFPDS	Time-Dependent Pollution-Routing Problem of Free Pickup and Delivery of passengers to the airport Service

TDPRP	Time-Dependent Pollution-Routing Problem
TDVRP	Time-Dependent Vehicle Routing Problem
TDVRP-PF	Time-Dependent Vehicle Routing Problem with Path Flexibility
TDVRPTW	Time-Dependent Vehicle Routing Problem with Time Windows
TD-VRSP-CO2	Time-Dependent Vehicle Routing & Scheduling Problem with CO2 emissions optimization
TS	Tabu Search
UK	United Kingdom
VRP	Vehicle Routing Problem
VRPPD	Vehicle Routing Problem with Simultaneous Pickup and Delivery
VRPTW	Vehicle Routing Problem with Time Windows
VUC	*Veículo Utilitário de Carga* (used as a synonym for LDT in our research)
ZMRC	*Zona de Máxima Restrição de Circulação* (Zone of Maximum Traffic Restrictions).

Table of Contents

1 Introduction

Green Logistics is a branch of Operations Research that has recently received increasing and close attention from governments and business organizations (Dekker et al., 2012; Jaehn, 2016; Lin et al., 2014). As defined by Dekker et al. (2012), green logistics is the "study of practices that aim to reduce the environmental externalities, mainly related to greenhouse gas emissions, noise and accidents, of logistics operations and therefore develop a sustainable balance between economic, environmental and social objectives". It involves all aspects of logistics, i.e., transportation, warehousing and inventories, and is motivated by the fact that the world's current production and distribution strategies are not sustainable in the long term.

Among the major logistics areas and with respect to the environment, transportation is the most visible aspect of supply chains as it accounts for large shares of the emissions of CO_2 and other greenhouse gasses (GHG), as well as other pollutants such as NO_x, SO_2 and particulate matters (e.g., fine dust and soot) (Bektas and Laporte, 2011; Dekker et al., 2012; Demir et al., 2014a; Jaehn, 2016). According to Demir et al. (2014), significant amounts of pollutants are emitted by trucks running on diesel engines, which accounts for a significant portion of freight transportation.

In this way, environmental aspects can be integrated into vehicle routing problems (VRP) to focus on minimizing the emission of pollutants or the vehicles' fuel consumption, instead of purely economic goals like total traveled distances, traveled times or the required number of vehicles (Jaehn, 2016). This kind of problem was first introduced by Palmer (2007), in which the author investigates the role of speed in reducing CO_2 emissions under various congestion scenarios and time window settings and proposes an integrated routing and emissions model. Since that year, several VRP variants concerning green aspects have been increasingly published in the literature (M. Figliozzi, 2010; Kuo, 2010; Xiao et al., 2012). The review by Lin et al. (2014) presents a wide mapping of green VRPs. However, as reminded by Jaehn (2016), Green VRPs must not be confused with the term GVRP, often referred to the specific VRP variant Green Vehicle Routing Problem (Erdoğan and Miller-Hooks, 2012), that employs an alternative fuel-powered fleet, in which vehicles must stop at refueling or recharging stations.

Also, in environments with high congestion such as large urban centers, not considering the urban traffic in routing decisions can lead to non-optimal solutions to the problem. This occurs

mainly because the travel time to traverse a road also depends on the departure time of the vehicle, not only the traveled distance (Franceschetti et al., 2017, 2013; Ichoua et al., 2003; Malandraki and Daskin, 1992). Thus, to surpass this limitation and to obtain more realistic solutions when concerning congestion and traffic conditions, time-dependent vehicle routing problems (TDVRP) have been proposed in the literature, such as the seminal work by Malandraki and Daskin (1992). According to Figliozzi (2012), poorly designed delivery routes leading vehicles to congested areas not only increase costs but also worsen some externalities, such as GHG emissions, noise and accidents. The paper by Gendreau et al. (2015) presents a literature review on the different classes of time-dependent problems in vehicle routing. Variants of time-dependent problems considering environmental aspects have also been addressed in the literature, such as works by Figliozzi (2011), Jabali et al. (2012), Qian and Eglese (2016) or Norouzi et al. (2017).

The Pollution-Routing Problem (PRP), proposed by Bektas and Laporte (2011) as an extension of the vehicle routing problem with time windows (VRPTW), is a VRP problem with a broader and more comprehensive objective function, that accounts for the amount of GHG emissions, fuel consumption, travel times and speeds and a variety of costs, not only related to the traveled distances. The PRP also involves the optimization of traveled speeds in each arc, as the vehicle's speed directly impacts the amounts of emissions. The authors also highlight the importance of speed as a decision variable in terms of producing energy-efficient solutions, noting that it also plays an important role when time windows are imposed. Demir et al. (2012) propose an Adaptive Large Neighborhood Search to solve the PRP.

Variants of the PRP have been proposed, such as a bi-objective version (Demir et al., 2014b), the time-dependent pollution-routing problem (TDPRP) (Franceschetti et al., 2017, 2013), and a practical version of the problem (PPRP) (Suzuki, 2016) that employs much simpler computations to estimate vehicles' fuel consumption, based on the Fuel Consumption Rate (FCR) by Xiao et al. (2012). Using Bektaş et al. (2016) notation, we refer to all of these problems and many others (e.g., Figliozzi, 2010; Xiao and Konak, 2016) simply as different kinds of PRPs.

Several methods are used to model the fuel consumption of the vehicles or the amounts of emitted CO_2 and other pollutants. Demir et al. (2014a) present a vast literature review of these models that can be classified as a factor, a macroscopic or a microscopic model, which are given in increasing order of complexity. Xiao et al. (2012) and Suzuki (2016), for instance, use the FCR factor model. Other examples include the MEET (Methodology for calculating

transportation emissions and energy consumption) (Hickman et al., 1999) macroscopic model, used by Jabali et al. (2012) and Figliozzi (2010). Microscopic models are the instantaneous ones such as the comprehensive modal emission model (CMEM) (Barth et al., 2005), widely used in the PRP literature (Bektas and Laporte, 2011; Demir et al., 2014b; Franceschetti et al., 2013; Huang et al., 2017). These fuel consumption models will be properly reviewed in the next section.

However, except for some factor models, most macroscopic and microscopic models employ significant numbers of data and parameters to provide the estimations. And despite the literature on PRPs and Green VRPs being extensive (see, e.g., Bektaş et al., 2016; Demir et al., 2014a; Lin et al., 2014), works focused on practical versions of time-dependent green VRPs are still scarce. In fact, to our knowledge the first mentioning to the term Practical Pollution-Routing Problem (PPRP) appears in Suzuki (2016), stating that from users' point of view, comprehensive models (e.g., the CMEM), while precise, may have limited practical values as they are too complex and require many parameters and user inputs. The author also points out suggestions to overcome some limitations of their problem, such as incorporating the time-dependent nature of the vehicle speed to the PPRP to account for the effects of congestion and traffic conditions, for example. In addition, we weren't able to find any work in the literature that follows Suzuki (2016)'s or a similar approach, thus identifying this gap of research.

All things considered, our research makes the following contributions: (1) introduces the Practical Pollution-Routing Problem with Time-Dependent Speeds (PPRP-TD) and its mathematical modeling; (2) proposes benchmark instances based on real data from the logistics operations of a major retail company located in São Paulo, Brazil; and (3) presents solution methods based on heuristics and metaheuristics to obtain good results in practical computational times, applicable to the context of delivery planning for physical distribution of products concerning the reduction of fuel consumption.

The remaining of this research is organized as follows: a brief literature review on fuel consumption and emissions modeling, as well as a systematic literature on green vehicle routing, are presented in Section 2; in Section 3 we introduce the problem and its characteristics; a new mathematical formulation for the PPRP-TD is given in Section 4; in Section 5 we describe our proposed solution methods to both the PPRP and PPRP-TD; Section 6 details our experiments and shed some insights on the problems and time-dependent freight transportation in São Paulo; and finally, in Section 7 we present our conclusions and suggestions for future research.

2 Literature Review

A brief literature review on fuels and emission modeling is presented in this section, along with a systematic review on the recent research in green vehicle routing, "an area of research that aims to reduce environmental externalities of transportation by better planning" (Bektaş et al., 2016). Similarly to Bektaş et al. (2016), we mainly focused on approaches based on traditional freight transportation (e.g., diesel trucks), meaning that new technologies such as alternative fuel-powered vehicles (e.g., electric and hybrids vehicles) or unmanned aerial vehicles were not covered.

2.1 Modeling of Emissions and Fuel Consumption

CO_2 emissions in road transportation are directly related to the fuel consumption of vehicles. In this way, several studies use fuel consumption as proxies to reduce GHG emission. However, there are models focused on the estimate of GHG emissions, models on fuel consumption and mixed models. These can be categorized into three groups of increasing levels of complexity: factor models, macroscopic models and microscopic models (Demir et al., 2014a). Factor models use simple fuel consumption methods, which do not employ aggregate network parameters to estimate network-wide emission rates like macroscopic models do. On the other hand, microscopic models are much more detailed and accurate, where instantaneous vehicle fuel consumption and emission rates are estimated (Demir et al., 2014a).

In this way, emissions and fuel consumption can be affected by various factors. These factors have been widely studied in the literature, such as works by Bigazzi and Bertini (2009), Demir et al. (2011), Demir et al. (2014a), Suzuki (2016) and Paschoal et al. (2017), among others. These factors range from the vehicle (e.g., its curb weight, shape, age and engine) and environment (e.g., road gradient and altitude) related factors to others such as traffic (e.g., speed and congestion), driver and operations (e.g., fleet size and mix, payload). Figure 1, by Demir et al. (2014a), summarizes most of these factors affecting emissions and fuel consumption.

Demir et al. (2011) presented a comparative analysis of several vehicle emission models for road freight transport. The following four microscopic and two macroscopic models are

compared: (i) an instantaneous fuel consumption model (IFCM) (Bowyer et al., 1985); (ii) a four-mode elemental fuel consumption model (FMEFCM) (Bowyer et al., 1985); (iii) a running speed fuel consumption model (RSFCM) (Bowyer et al., 1985); (iv) a comprehensive modal emission model (CMEM) (Barth et al., 2005); (v) the methodology for calculating transportation emissions and energy consumption (MEET) (Hickman et al., 1999), macroscopic model; (vi) the Computer *programme* to calculate emissions from road transportation (COPERT) (Kouridis et al., 2010), macroscopic model. The authors performed various simulations using broadly realistic assumptions and concluded that results are overall consistent with expectations (e.g., responses to variations of the size of the vehicle, speed or road gradient), though the models present some variance in results.

Paschoal et al. (2013) used a similar approach, comparing four of those six compared by Demir et al. (2011) in a field study in Brazil. They analyzed a route from a factory in the city of Itu (SP) to its depot in São Paulo, comparing the actual vehicle fuel consumption of the route to those estimated by the four used fuel consumption models: (i) IFCMC; (ii) FMEFCM; (iii) RSFCM; and (iv) CMEM. Their empirical results indicate the RSFCM had the best estimate results regarding consumed fuel and the addressed conditions. Demir et al. (2011) also pointed out that these models can vary in performance when comparing modeled results with actual road use data.

Besides MEET and COPERT, a couple of other available macroscopic models are: (a) National Atmospheric Emissions Inventory (NAEI) (NAEI, 2015); and (b) EcoTransIT (Heidelberg et al., 2014).

In the following subsections, we detail some of the used macroscopic models and the CMEM microscopic model. It is worth noting that the intent here is not to present an extensive review of available models but to show how emissions and fuel consumption are often modeled and estimated and which factors are used in these models. For more detailed reviews we refer the interested reader to the works by Demir et al. (2014a), Bektaş et al. (2016) and Cunha et al. (2017).

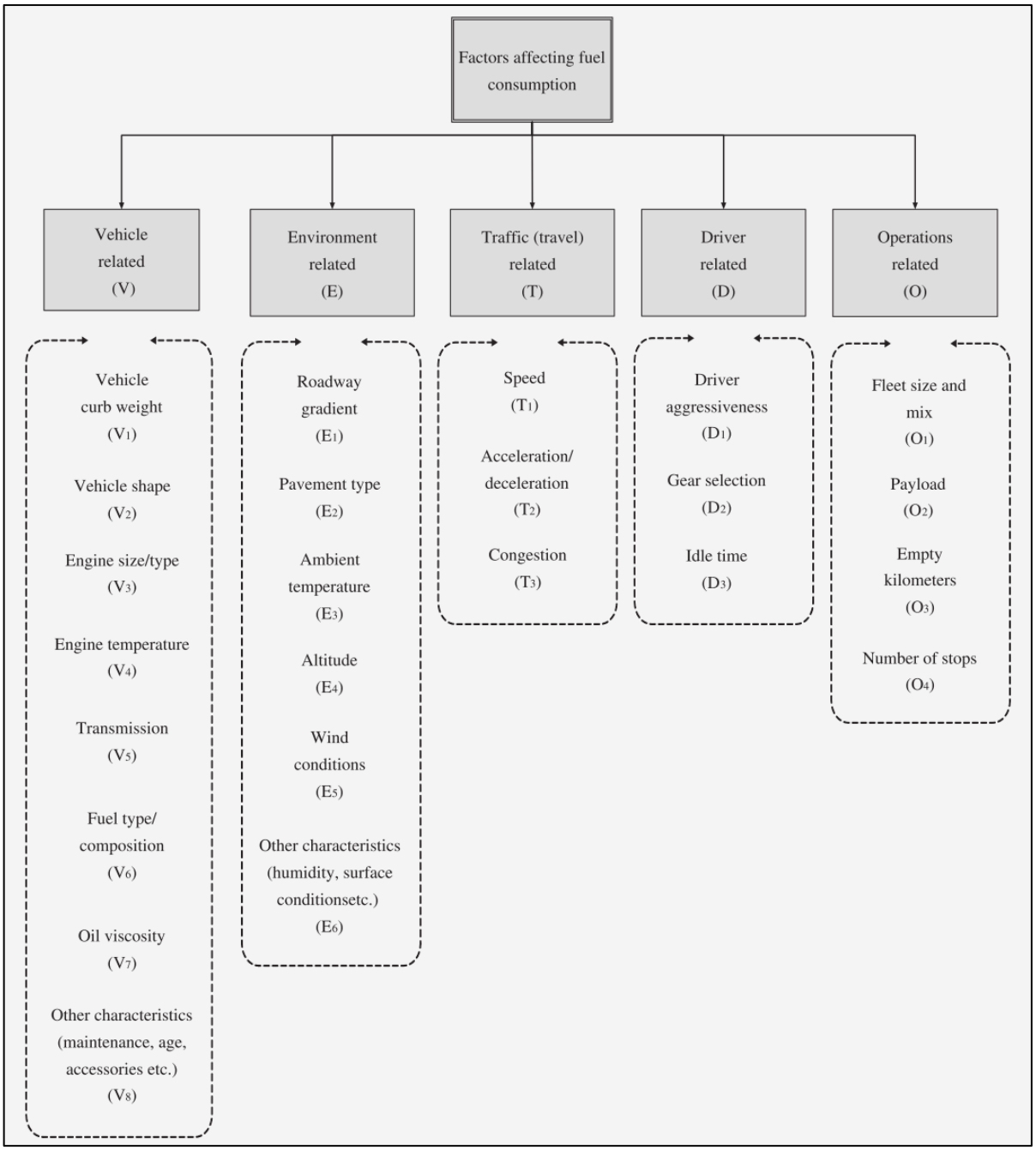

Figure 1 – Factors affecting fuel consumption (Demir et al., 2014a).

2.1.1 Macroscopic models

2.1.1.1 Methodology for calculating transportation emissions and energy consumption (MEET)

The MEET project was first introduced by a publication of the European Commission (Hickman et al., 1999), to provide a methodology for calculating transportation emissions and energy consumption. The model is based on on-road measurements and parameters based on real-life experiments and covers several vehicle technologies for different types and classes of road vehicles, as well as rail, shipping, and air transport. In road transport, it addresses cold start extra emissions, evaporative losses, road gradient and vehicle load effects for heavy goods vehicles.

For diesel vehicles weighing less than 3.5 tons, CO_2 emissions are calculated using the speed-dependent regression function $e(v) = 0.0617v^2 - 7.8227v + 429.51$. For larger classes of vehicles, MEET indicates the use a function in the form $e(v) = K + av + bv^2 + cv^3 + d/v + e/v^2 + f/v^3$. In both functions, $e(v)$ is the rate of CO_2 emissions in grams/kilometer and v is the mean speed of the vehicle. Values for parameters K and a to f are presented in Table 1.

According to Hickman et al. (1999), other parameters also affect emissions directly or by altering the engine's operation mode (e.g., road gradient and vehicle load). However, the previous functions consider an unloaded vehicle on a road with zero gradients and zero altitudes (i.e., standard testing conditions). To account for these kinds of factors on emissions, MEET suggests some correction functions. The road gradient correction factor $GC(v) = A_6v^6 + A_5v^5 + A_4v^4 + A_3v^3 + A_2v^2 + A_1v + A_0$ is used to take the effect of road gradient into account, where v is the mean speed of the vehicle and coefficients A_0 to A_6 are pollutant and vehicle specific, which values for CO_2 emissions can be found in Table 2.

To take the load factor into account, MEET uses the following load correction function: $LC(\gamma, v) = k + n\gamma + p\gamma^2 + q\gamma^3 + rv + sv^2 + tv^3 + u/v$, where v is the mean velocity of the vehicle in km/h, γ is the gradient in percent, k is a constant and n to u are coefficients, found in Table 3.

In this way, the total CO_2 emissions (gram) is estimated by MEET using Expression (1), where v is the mean speed, γ is the road gradient and D is the traveled distance.

$$E(\gamma, v, D) = e(v) \cdot GC(v) \cdot LC(\gamma, v) \cdot D \quad (1)$$

Different coefficients values for other vehicle, pollutants and road gradients, as well as ambient temperature correction functions and other features, are available in Hickman et al. (1999). However, as pointed out by Bektaş et al. (2016), all of these model parameters were conceived and calibrated in 1999, which calls for updates as new engine technologies have been since developed. In this way, the MEET model should be used with caution while applied to modern trucks.

Table 1 – Coefficients of CO_2 emission functions used in MEET.

Weight class	K	a	b	c	d	e	f
$3.5t <$ Weight $\leq 7.5t$	110	0	0	0.000375	8702	0	0
$7.5t <$ Weight $\leq 16t$	871	-16.0	0.143	0	0	32031	0
$16t <$ Weight $\leq 32t$	765	-7.04	0	0.000632	8334	0	0
Weight $> 32t$	1576	-17.6	0	0.00117	0	36067	0

Table 2 – Coefficients of gradient factor function for MEET.

Weight class	A_6	A_5	A_4	A_3	A_2	A_1	A_0	**Slope (%)**
$3.5t <$ Weight $\leq 7.5t$	0	-3.01E-09	5.73E-07	-4.13E-05	1.13E-03	8.13E-03	9.14E-01	[0,4]
$3.5t <$ Weight $\leq 7.5t$	0	-1.39E-10	5.03E-08	-4.18E-06	1.95E-05	3.68E-03	6.69E-01	$[-4,0]$
$7.5t <$ Weight $\leq 16t$	0	-9.78E-10	-2.01E-09	1.91E-05	-1.63E-03	5.91E-02	7.70E-01	[0,4]
$7.5t <$ Weight $\leq 16t$	0	-6.04E-11	-2.36E-08	7.76E-06	-6.83E-04	1.79E-02	6.12E-01	$[-4,0]$
$16t <$ Weight $\leq 32t$	0	-5.25E-09	9.93E-07	-6.74E-05	2.06E-03	-1.96E-02	1.45E+00	[0,4]
$16t <$ Weight $\leq 32t$	0	-8.24E-11	2.91E-08	-2.58E-06	5.76E-05	-4.74E-03	8.55E-01	$[-4,0]$
Weight $> 32t$	0	-2.04E-09	4.35E-07	-3.69E-05	1.69E-03	-3.16E-02	1.77E+00	[0,4]
Weight $> 32t$	0	-1.10E-09	2.69E-07	-2.38E-05	9.51E-04	-2.24E-02	9.16E-01	$[-4,0]$

Table 3 – Coefficients of the load correction function for MEET.

Weight class	k	n	p	q	r	s	t	u
$3.5t <$ Weight $\leq 7.5t$	1.27	0.0614	0	-0.00110	-0.00235	0	0	-1.33
$7.5t <$ Weight $\leq 16t$	1.26	0.0790	0	-0.00109	0	0	-2.03E-7	-1.14
$16t <$ Weight $\leq 32t$	1.27	0.0882	0	-0.00101	0	0	0	-0.483
Weight $> 32t$	1.43	0.121	0	-0.00125	0	0	0	-0.916

2.1.1.2 Computer *programme* to calculate emissions from road transportation (COPERT)

The COPERT is another model that can be used to calculate vehicle emissions of several pollutants and GHGs in road transportation, for a range of vehicles by engine classification and vehicle type. The development of the COPERT model was funded by the European Economic Area, and its first version dates from 1989. The current and updated version of COPERT 4 can be found in Kouridis et al. (2010).

Like MEET, COPERT uses many regression functions to estimate fuel consumption, based on the vehicle class, engine technology, and speed. Expression (2) shows an example of a total fuel consumption function (gram) with different load and gradient factors, where v is the average speed of the vehicle and D is the traveled distance. Coefficients a to e are given in Table 4.

$$F(v, D) = (e + (a \exp(-bv)) + (c \exp(-dv))) \cdot D \quad (2)$$

Table 4 – Set of sample data (i.e., for a 20–26t rigid truck running on Euro 5 diesel) to be used with COPERT (Bektaş et al., 2016).

Payload (%)	**Gradient (%)**	***a***	***b***	***c***	***d***	***e***
0	0	530.707	0.0634	2704.528	0.512	157.588
0	-2	546.477	0.064	9599.652	0.766	61.960
0	+2	1051.552	0.424	-67.688	0.084	0
50	0	505.770	0.051	4762.796	0.609	180.436
50	-2	479.620	0.047	7858.071	0.677	40.246
50	+2	2074.874	1.008	-0.534	0	0
100	0	502.941	0.041	9343.090	0.729	195.202
100	-2	1144.824	0.981	-0.400	0	0
100	+2	1883.813	1.006	-0.422	0	0

2.1.2 Microscopic model

2.1.2.1 Comprehensive modal emission model (CMEM)

The CMEM introduced by c and Barth and Boriboonsomsin (2009) presents a heavy-duty truck fuel consumption and emission model with several sub-models, each for a specific category of vehicle and technology. Developed models use an approach where the emission process is divided in modules, each corresponding to a physical phenomenon associated with the vehicle operation and emission (e.g., idle, steady-state, cruise, acceleration and deceleration). The CMEM is based on instantaneous tailpipe emissions data collected from 343 light-duty vehicles tested for various driving cycles (Demir et al., 2014a).

Barth et al. (2005) mention that existing macroscopic models are good for predicting emission inventories for large regional areas but lack the precision to estimate emissions and fuel consumption at a more detailed and microscopic level (e.g., with second-by-second measures, ramp metering, signal coordination, and other Intelligent Transportation Systems methods).

Despite its accurate estimations, the model requires a number of user inputs and detailed vehicle parameters, such as the engine friction coefficient, drag, and rolling resistance coefficients or the engine speed and displacement. CMEM contains the following three modules:

(i) The engine power module: represents the total tractive power demand $P_{tract}(t)$ (kilowatt) required to the vehicle to move, given by Expression (3), where M is the total mass (kg) of the vehicle, v is speed (m/s), a is the acceleration (m/s²), g is the gravitational constant (9.81 m/s²), θ is the road angle, A is the frontal surface area of the vehicle (m²), ρ is the air density (kg/m³), and C_r and C_d are the rolling and drag resistant coefficients, respectively. To convert the tractive demand into engine power requirement based on the vehicle drive train efficiency η_{tf}, it is used Expression (4), where $P(t)$ represents the second-by-second engine power output (kilowatt) and P_{acc} are engine power demand associated with the operation of accessories, such as air conditioning, power steering and brakes, and electrical loads.

$$P_{tract}(t) = (Ma = Mg \sin\omega(\theta) + 0.5 C_d \rho A v^2 + MgC_r \cdot \cos\omega(\theta)) \frac{v}{1000} \quad (3)$$

$$P(t) = \frac{P_{tract}(t)}{\eta_{tf}} + P_{acc} \quad (4)$$

(ii) The engine speed module: the engine speed $N(v)$ (in rpm) is calculated in function of the vehicle speed v using Expression (5), where S is the engine-speed/vehicle-speed ration in top gear L_g and $R(L)$ is the gear ratio in gear $L = 1, \dots, L_g$.

$$N(v) = S\frac{R(L)}{R(L_g)}v \tag{5}$$

(iii) The fuel rate module: the fuel rate $f_{cm}(t)$ (in gram/second) is given by Expression (6), where k is the engine friction coefficient, V is the engine cubic capacity (liter), η is the efficienty parameter for diesel engines and $\xi = (1 + b_1(N(v) - N_0)$. In this last expression, $b_1 = 10^{-4}$ is an adjust coefficient (Barth et al., 2005) and $N_0 \approx 30\sqrt{\frac{3.0}{V}}$ is the slow running engine speed (rpm).

$$f_{cm}(t) = \xi\left(kN(v)V + \frac{P(t)}{\eta}\right)\cdot\frac{1}{43.2} \tag{6}$$

In this way, the total fuel consumption (gram) is calculated using Expression (7).

$$F_{cm}(T) = \int_0^T f_{cm}(t)dt \tag{7}$$

Demir et al. (2014a) mention that the CMEM could be seen as the current state-of-the-art in microscopic emission modeling, because of its accuracy and ease of application.

Recently, however, Turkensteen (2017) mentioned that one main challenge associated with the use of microscopic models (e.g., CMEM) to obtain accurate total emissions is that the user has to input all of these previously mentioned parameters and variables on a moment-to-moment basis, providing instantaneous emissions. In other words, these models may not be practical to be implemented by practitioners. In fact, the author points out that many vehicle routing problems in the literature (e.g., Bektas and Laporte, 2011) assume fixed speed computations, which is unlikely to happen for long periods of time (specially in an urban environment, as traffic conditions force the driver to adopt lower or higher speeds and sometimes vehicles can even stop), thus simplifying and adapting the CMEM model. Turkensteen (2017) provides some interesting insights, indicating that fixed speed computations might not be sufficiently accurate to be used in green routing problems, based on the analysis of a series of different driving cycles. Fixing this kind of situations, however, leads to increasing the complexity of the model eventually.

2.2 Green Vehicle Routing

Green vehicle routing is a branch of green logistics that has recently received increasing and close attention from governments and business organizations. It refers to vehicle routing problems that explicitly consider the externalities associated to the use of vehicles that are not sustainable in the long term (e.g., CO_2 and other GHG emissions) (Lin et al., 2014). In other words, Green Vehicle Routing Problems (Green VRPs) are problems in which the main concern is to balance environmental and economic costs, by implementing effective routes to reduce the environmental impact of vehicle routes, such as the fuel consumption or the emissions of GHGs directly.

In the VRP literature, GVRP can also refer to works that employ the use of alternative fuel-powered vehicles (AFVs, such as electric vehicles), which usually need to make stops at recharging or refueling stations during the routes (Erdoğan and Miller-Hooks, 2012). However, these kinds of problems are not covered in our review, as the focus here is on traditional road transportation. For recent reviews concerning electric vehicles, interested readers can refer to the works by Pelletier et al. (2016) and Juan et al. (2016). VRPs in reverse logistics are also often included in the Green VRP literature, because of their environmental-friendly aspect. These works cover selective pickups and pricing, waste management, end-of-life goods collection and simultaneous distribution and collection (Lin et al., 2014), but are also not in the extension of our review as they do not focus on the modeling of emissions or fuel consumption.

Lin et al. (2014) surveyed and reviewed about 280 Green VRP works, focusing mostly on papers ranging from 2006 to 2012. Their search was conducted both in horizontal and vertical dimensions. In other words, first they paid attention to finding VRPs with sustainability aspects and their evolution on the timeline; and then they classified which variant of the VRP is used in each article.

In order to update Lin et al. (2014)'s survey, we conducted a systematic literature review on Green VRP. Journal articles, books, and conference papers were searched in three scientific databases: Web of Science, Scopus and Emerald Insight. The main used term was *(vehicle routing")AND((green)OR(emission*)OR(pollution))*, being searched in the title, abstract and keyword fields of the documents. Initial searches retrieved more than 400 works in total, after excluding duplicates (i.e., works that appeared in more than one database). Non-related articles were also discarded, as well as the already mentioned papers based on green vehicles (i.e.,

AFVs) and reverse logistics. This literature review focused mainly on articles published in periodic journals. In total, our review included 137 papers ranging from 2006 to 2018, in which 120 were published after 2012. Table 5 presents which journals had most published papers in our review, and Figure 2 shows the timeline evolution of published papers in the area.

Interestingly, in a recent review and state-of-the-art classification on VRPs by Braekers et al. (2016) that follows the taxonomy introduced by Eksioglu et al. (2009), no mention whatsoever was made to green vehicle routing nor to any emissions or sustainability aspects, which also calls for an update to include sustainability factors in VRP literature taxonomy reviews.

Among the reviewed papers, we identified several different time-dependent problems. Table 6 presents the used notation for these problems and Table 7 shows the characteristics for each identified time-dependent work.

Table 5 – Journals with most published papers in the review.

Journal[1]	**# Papers**	**IF JCR**[2]	**SJR**[3] **Indicator**	**Total Citations**[4]
Transportation Research Part D: Transport and Environment	12	2,341	1,195	355
European Journal of Operational Research	9	3,297	2,505	295
Expert Systems with Applications	7	3,928	1,433	278
International Journal of Production Economics	5	3,493	2,216	183
Transportation Research Part B: Methodological	5	3,769	2,742	370
Computers & Industrial Engineering	4	2,623	1,542	109
Annals of Operations Research	3	1,709	1,009	52
Journal of Cleaner Production	3	5,715	1,615	9
Sustainability	3	1,789	0,524	6
Transportation Research Part C: Emerging Technologies	3	3,805	1,935	78
Transportation Research Part E: Logistics and Transportation Review	3	2,974	1,694	313
Transportation Science	3	3,275	2,567	99
Applied Soft Computing Journal	2	3,541	1,308	33
Computers and Operations Research	2	2,6	2,326	200
IEEE Transactions on Intelligent Transportation Systems	2	3,724	1,018	30
International Transactions in Operational Research	2	1,745	1,01	7
Mathematical Problems in Engineering	2	0,802	0,277	10
Omega	2	4,029	3,674	79
Transportation Research Record	2	0,598	0,494	120
Other Journals (1 published paper each)	38			803

[1] Showing only JCR and SJR indexed journals. The total number of reviewed papers is 137.

[2] Journal Impact Factor 2016 - Journal Citation Reports (JCR).

[3] SCImago Journal Rank (SJR) indicator 2016.

[4] Total citations from Scopus and/or Web of Science in July 2017.

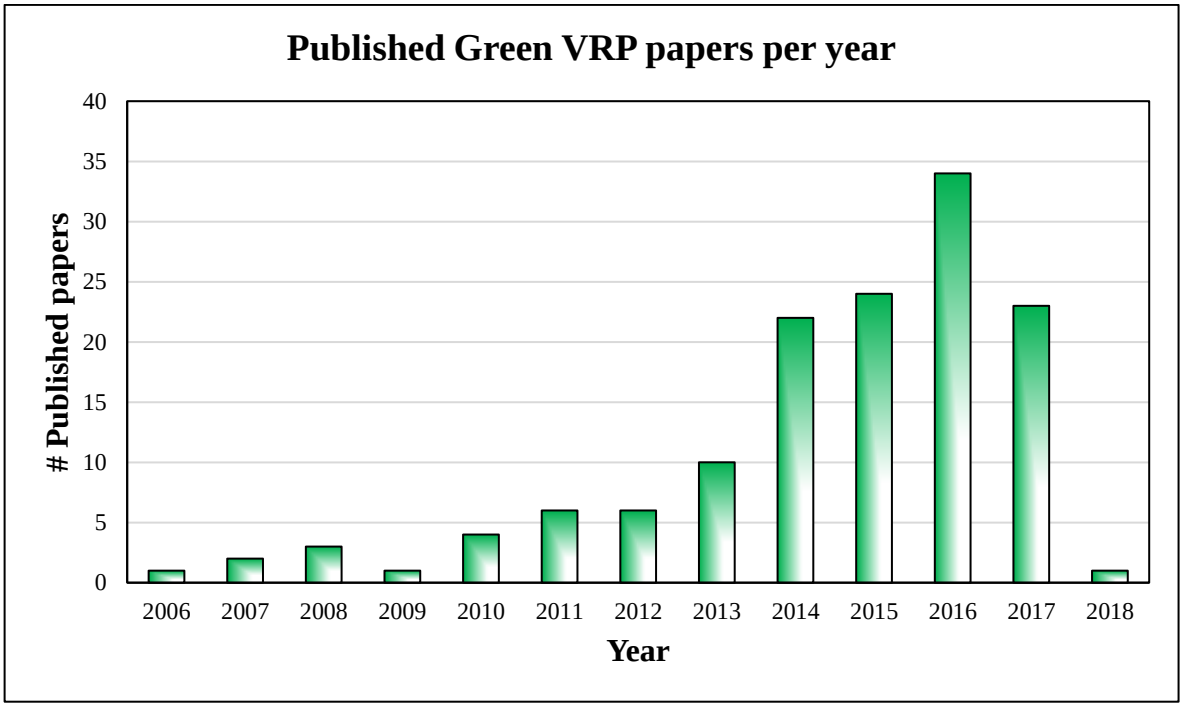

Figure 2 – Evolution of published papers included in our review.

Table 6 – Notation associated with cited problems in Table 7.

Notation	Problem
2E-CVRP	Two-Echelon Capacitated Vehicle Routing Problem
BTL-VRPTW	Bi-objective Time, Load and path-dependent Vehicle Routing Problem with Time Windows
E-TDVRP	Emissions-based Time-Dependent Vehicle Routing Problem
EVRP	Vehicle Routing Problem for Emissions minimization
Green STDCVRP	Green Stochastic Time-Dependent Capacitated Vehicle Routing Problem
Green TDCVRP	Green Time-Dependent Capacitated Vehicle Routing Problem
GVRSP	Green Vehicle Routing and Scheduling Problem
MTHVRPP	Minimal-carbon-footprint Time-dependent Heterogeneous-fleet Vehicle Routing Problem with alternative Paths
OTDVRP	Open Time Dependent Vehicle Routing Problem
TDFPDS	Time-Dependent Pollution-Routing Problem of Free Pickup and Delivery of passengers to the airport Service
TDPRP	Time-Dependent Pollution-Routing Problem
TDVRP	Time-Dependent Vehicle Routing Problem
TDVRP-PF	Time-Dependent Vehicle Routing Problem with Path Flexibility
TDVRPTW	Time-Dependent Vehicle Routing Problem with Time Windows
TD-VRSP-CO2	Time-Dependent Vehicle Routing & Scheduling Problem with CO2 emissions optimization
VRPTW	Vehicle Routing Problem with Time Windows

Table 7 – Summary of time-dependent problems in green vehicle routing.

Year	Work	Problem	Math. Model	Case Study	Solution Method
2017	(Xiao and Konak, 2017)	TD-VRSP-CO2	●		Exact Dynamic Programming / Genetic Algorithm
	(Androutsopoulos and Zografos, 2017)	BTL-VRPTW	●		Network Reduction / Ant Colony System
	(Çimen and Soysal, 2017)	Green STDCVRP			Approximate Dynamic Programming
	(Franceschetti et al., 2017)	TDPRP			Adaptive Large Neighborhood Search / Departure Time and Speed Optimization Procedure
	(Guo and Liu, 2017)	TDFPDS	●		Set-partitioning formulation / Departure Time and Speed Optimization Procedure
	(Huang et al., 2017)	TDVRP-PF	●	Beijing, China	
	(Norouzi et al., 2017)	TDVRP	●		Particle Swarm Optimization
	(Soysal and Çimen, 2017)	Green TDCVRP		Turkey	Simulation / Dynamic Programming
2016	(Alinaghian and Naderipour, 2016)	TDVRP		Esfahan, Iran	Gaussian Firefly Algorithm
	(Ehmke et al., 2016)	TDVRP	●	Stuttgart, Germany	Adapted Tabu Search
	(Naderipour and Alinaghian, 2016)	OTDVRP			Particle Swarm Optimization
	(Qian and Eglese, 2016)	VRPTW	●	London, England	Column generation based Tabu search
2015	(Soysal et al., 2015)	2E-CVRP	●	Netherlands	
	(Wen and Eglese, 2015)	TDVRPTW	●	London, England	LANCOST heuristic
	(Xiao and Konak, 2015)	GVRSP	●		Simulated Annealing
	(Yao et al., 2015)	TDVRPTW	●	Beijing, China	Ant Colony System
2014	(Liu et al., 2014)	MTHVRPP	●		Genetic Algorithm
	(Qian and Eglese, 2014)	TDVRP	●	Bristol, England	Dynamic Programming / New Heuristic Approach
2013	(Franceschetti et al., 2013)	TDPRP	●		MILP / Departure Time and Speed Optimization Procedure
2012	(Jabali et al., 2012)	E-TDVRP	●		Tabu Search
	(Saberi and Verbas, 2012)	EVRP	●		Continuous Approximation
2011	(Figliozzi, 2011)	TDVRP	●	Portland, OR	
2010	(Kuo, 2010)	TDVRP			Simulated Annealing
	(Maden et al., 2010)	VRPTW	●	United Kingdom (South West)	LANTIME

One of the first papers that address time-dependent speeds in a Green VRP is the one by Kuo (2010). The author proposes a model for calculating total fuel consumption for the time-dependent vehicle routing problem (TDVRP), in which speed and travel times are assumed to depend on the time of travel when planning vehicle routing. A simulated annealing (SA) algorithm is also proposed to solve the problem efficiently.

Maden et al. (2010) propose a tabu-search algorithm called LANTIME, which constructs vehicle routes that aim to minimize total travel time in a VRPTW. The method relies on data from a Road Timetable (Eglese et al., 2006) and tends to produce routes that avoid congestion, thus providing environmental benefits. The authors also present a case study where the LANTIME algorithm was applied using real data for a vehicle fleet from the South West region of the United Kingdom, analyzing the effects of using Road Timetable data compared with routing and scheduling where this information is not available.

Figliozzi (2011) analyses the impacts of congestion on time-definitive urban freight distribution networks CO_2 emission levels. The author presents results from a case study in Portland, Oregon, that uses travel time data from an extensive archive of freeway sensors. A TDVRP MILP model is presented to design commercial vehicle routes, which considers networks with time-dependent travel speeds, hard time windows, and real-world time/distance data (Figliozzi, 2011). Emissions are modeled using MEET and the research assumes that CO_2 emission levels are not significantly affected by cargo weight when compared to the impacts of travel speeds (Figliozzi, 2011). The TDVRP is solved using an algorithm called Iterated Route Construction and Improvement (IRCI), described in detail in Figliozzi (2010).

As previously mentioned, the Pollution-Routing Problem (PRP) was introduced by Bektas and Laporte (2011) as an extension of the VRPTW with a more comprehensive objective function that minimizes labor, fuel and emission costs, expressed as a function of load, speed and other parameters. A key aspect of the PRP is that it also involves the optimization of traveled speeds in each arc, as the vehicle's speed directly impacts the amounts of emissions. The authors also highlight the importance of the speed as a decision variable in terms of producing energy-efficient solutions, emphasizing that it plays an important role when time windows are in place. To illustrate the problem, Bektas and Laporte (2011) generated several instances with up to 20 nodes, each node representing cities from the United Kingdom (UK).

To solve the PRP, Demir et al. (2012) propose an Adaptive Large Neighborhood Search (ALNS) metaheuristic, which is integrated with a specialized speed optimization procedure (SOP). The ALNS uses fixed speeds as inputs and then the SOP is run on each route to

improving the solution, computing optimal speeds on the given paths in order to minimize fuel consumption, emissions and driver costs. The authors also generated instances (with sizes ranging from 10 to 200 nodes) representing randomly selected cities from the UK. The efficiency of the algorithm was confirmed by the results of extensive computational experiments.

In addition, variants of the PRP have also been proposed. Demir et al. (2014b) present the bi-objective Pollution-Routing Problem, in which conflicting objective functions (i.e., the minimization of fuel consumption and driving times) are considered separately. The bi-objective PRP is solved using the same ALNS with SOP described in Demir et al. (2012). The algorithm is used to find and store the non-dominated solutions and then four *a posteriori* bi-objective solution methods are tested, being evaluated by means of two performance indicators (i.e., the hypervolume and epsilon indicators). Again, instances representing cities from the UK are used in the experiments.

The time-dependent pollution-routing problem (TDPRP) (Franceschetti et al., 2017, 2013), for instance, is another variant of the PRP. It considers traffic congestion that significantly affects vehicle speeds and increases emissions during peak periods. Franceschetti et al. (2013) introduced the problem, describing a mixed integer linear programming (MILP) formulation and providing insights about the tradeoffs involved. The authors also present several examples that motivate idle waiting time at customer nodes (either before or after service times) to minimize a total cost objective function. The authors also describe a novel departure time and speed optimization procedure (DSOP) that can be used on a fixed route, build on the analytical results given for the single-arc version of the problem. To effectively solve large instances of the TDPRP, Franceschetti et al. (2017) propose an algorithm based on the ALNS metaheuristic that use the DSOP as a subroutine to optimize the departure times and vehicle speeds.

A simpler version of the PRP is presented by Suzuki (2016) in the form of the Practical Pollution-Routing Problem. Factors included in the model were obtained by expert inputs and opinions from managers of motor carriers from different sizes. The PPRP uses the Fuel Consumption Rate (FCR) approach, based on statistical data published by the Ministry of Land, Infrastructure, Transport, and Tourism of Japan, which indicates that the fuel consumption rate is strongly correlated to the vehicle's gross weight (Xiao et al., 2012). Suzuki (2016) propose a MILP formulation for the problem. To solve the PPRP, the author decomposes the objective function into two parts and solve it using a simulated annealing (SA) metaheuristic to

approximate a Pareto frontier. Then, a tabu search (TS) step is used to improve each element of the efficient frontier. Details on the PPRP and FCR are provided in the next section.

In the context of time-dependent vehicle routing, Jabali et al. (2012) have investigated travel times, fuel consumption and CO_2 emissions in a TDVRP. The authors propose a framework for modeling CO_2 emissions and also provide analysis in which congestion takes part, forcing vehicles to drive slower and therefore emitting more pollutants. The problem is solved via a tabu search procedure. Saberi and Verbas (2012) present a continuous approximation model for solving the vehicle routing problem for emissions minimization (EVRP), a variant of the TDVRP in which minimizing emissions is an additional objective of the problem. Both Jabali et al. (2012) and Saberi and Verbas (2012) use MEET to estimate the emissions.

Qian and Eglese (2014) consider a problem of finding routes and schedules with least fuel emission in a time-dependent network. In their model, speeds are also decision variables that are limited by the level of congestion on the roads. To solve the problem, the authors propose two solution methods: (i) a time-increment-based dynamic programming method; and (ii) a heuristic approach, which contains a route selection process and a speed adjustment process. Computational experiments with real traffic data from Bristol are presented, indicating that further savings in fuel emissions can be achieved in the time optimization model.

Liu et al. (2014), for instance, investigate the Minimal-carbon-footprint Time-dependent Heterogeneous-fleet Vehicle Routing Problem with alternative Paths (MTHVRPP) and propose a Genetic Algorithm to solve it. The MRHVRPP simultaneously considers different vehicle types and alternative path selections to increase its applicability to different situations, also considering energy consumption during different times of the day.

On the other hand, Soysal et al. (2015) consider the environmental aspects of multi-echelon distribution networks, in which freight is delivered via intermediate depots. The authors tackle the time-dependent Two-Echelon Capacitated Vehicle Routing Problem (2E-CVRP), presenting a comprehensive MILP formulation for the problem that considers the vehicle type, traveled distance, speed, load, emissions and different time zones. The authors also present a case study that shows the applicability of the model in a supermarket network in the Netherlands, concluding that environmentally friendly solutions can be achieved using a two-echelon network, in contrast to least-cost solutions found in single-echelon scenarios.

Wen and Eglese (2015) presented a heuristic algorithm called LANCOST, which is an improvement to the LANTIME heuristic (Maden et al., 2010). The LANCOST algorithm is

proposed to solve Time-Dependent Vehicle Routing Problems with Time Windows, in which speeds of the vehicles traveling along any road varies depending of the departure time of travel. The authors also design a benchmark dataset and also present a case study with real data from heavy-duty trucks (HDT) from a supermarket chain in London. Yao et al. (2015) also addressed a TDVRPTW considering minimizing fuel consumption, but takes hand of an Ant Colony Algorithm to solve it, while performing computational experiments carried out on the real network of Beijing, China. Xiao and Konak (2015), on the other hand, solved a green vehicle routing and scheduling problem (GVRSP) with general time-dependent traffic conditions and hierarchical objectives and weighted tardiness using a simulated algorithm.

By extending the research presented in Qian and Eglese (2014), in Qian and Eglese (2016) the authors propose a VRPTW in which the objective is to minimize the total emissions in terms of the amount of GHG emitted, measured in CO_2 equivalents. They present a column generation based tabu search algorithm to solve the problem, also testing it with real data from the London network. Ehmke et al. (2016) also aims to minimize CO_2 emissions in a TDVRP, but the authors test their adapted tabu search algorithm using real data from Stuttgart, Germany. In the same of research, Alinaghian and Naderipour (2016) solve a TDVRP with multi-alternative graph to reduce fuel consumption, using a Gaussian Firefly Algorithm and presenting a case study from Esfahan, Iran.

Huang et al. (2017) study the Time-Dependent Vehicle Routing Problem with Path Flexibility (TDVRP-PF), a problem that explicitly considers path selection in the road network as an integrated decision in the TDVRP. The authors formulate the problem under stochastic traffic conditions and test their Route-Path approximation solution method on a testbed of instances inspired on the real network of Beijing.

Çimen and Soysal (2017) also presented a stochastic approach to a TDVRP. The authors solve the Green Stochastic Time-Dependent Capacitated Vehicle Routing Problem (Green STDCVRP) using an Approximate Dynamic Programming based heuristic algorithm, which has Machine Learning and Neural Networks components. At an earlier stage of their research the authors also have addressed a Green TDCVRP that accounts for transportation emissions in the objective function in Soysal and Çimen (2017), but with no stochastic considerations. In this earlier research, the authors proposed a proposed a generic heuristic approach (i.e., a Simulation Based Restricted Dynamic Programming solution method) and test it in a real life urban distribution planning problem for a pharmaceutical warehouse owned by one of the largest pharmaceutical distribution companies in Turkey.

Xiao and Konak (2017) propose a genetic algorithm with exact dynamic programming (DP) for the time-dependent vehicle routing & scheduling problem with CO_2 emissions optimization (TD-VRSP-CO_2). The authors present an enhanced mixed integer linear programming (MILP) model for the problem, considering the impact of varying payload weight on the vehicle's emissions. For the scheduling part of the problem, an exact algorithm based on DP is proposed. In addition, a solution approach based on a hybrid genetic algorithm combined with the DP algorithm is used to yield efficient solutions to the problem.

A different problem is addressed by (Guo and Liu, 2017), in which the authors tackle a Time-Dependent Vehicle Routing of Free Pickup and Delivery Service (TDFPDS) that accoungs for carbon emissions. To solve the problem, they use an exact approach, based on a set partitioning formulation and the Departure Time and Speed Optimization Procedure (DSOP).

3 Problem Description

As previously mentioned, Suzuki (2016) proposed the Practical Pollution-Routing Problem (PPRP), noting that comprehensive microscopic models, while precise, may have limited practical values because of their complexity and numerous parameters. Since several factors affect fuel consumption (Demir et al., 2014a) and including them all (e.g., using a microscopic model such the CMEM) requires significant amounts of data (e.g., regarding the vehicles, the environment, the operation), practitioners are likely not to have all of these data at hand. These situations might lead the companies to adopt other purely economic objectives, such as total traveled time or distance instead of environmental-friendly decisions.

Suzuki (2016) conducted on-site interviews with decision makers from three different sized motor carriers: (i) a small private carrier, that owns roughly 30 trucks; (ii) a medium-sized for-hire carrier with roughly 300 trucks; and (iii) a large for-hire carrier that operates over 2000 trucks. Based on the managers' inputs and expert opinions from these carriers, they classified the factors to be included in their model. *"Drivers"* and *"fleet size"* are factors found to be of trivial importance, therefore were not included in the model. *"Payload"*, on the other hand, belongs to the category of factors that are included in the model as decision variables, while the *"speed"*, *"gradient"*, and *"congestion"* factors were all deemed important by carrier managers and thus are in the model, but are not treated as decision variables (Suzuki, 2016), such as in our case.

The PPRP uses the Fuel Consumption Rate (FCR) approach presented by Xiao et al. (2012), which is based on statistical data. By dividing the vehicle's combined gross weight into two parts (the vehicle's no-load, and the carried load), Xiao et al. (2012) formulated a linear FCR function in the form of Expression (8). ρ_{ijk} represents the fuel consumption rate (km/L) of a given vehicle k when traveling from a customer i to a customer j, y_{ijk} is the payload for the same arc (i,j) and vehicle k, Q is the vehicle capacity, and ρ_0 and ρ^* are the vehicle corresponding no-load FCR and full-load FCR, respectively.

$$\rho_{ijk} = \rho_0 + \frac{\rho^* - \rho_0}{Q} y_{ijk} \tag{8}$$

As data used by Xiao et al. (2012) derives from passenger cars running on gasoline, we use a different FCR function to estimate the lower and upper bound values of ρ_0 and ρ^*. Our

formulated FCR function is based on GPS data from light-duty trucks (i.e., VUC, from Portuguese *'veículo utilitário de carga'*) delivering in the São Paulo Metropolitan Region (RMSP) and modeled using the CMEM model; the process is detailed in Section 6. Other FCR functions from the literature are given for different classes of vehicles in Section 6 as well. Similarly to Suzuki (2016), Expression (8) indicates that ρ_{ijk} increases by $(\rho^* - \rho_0)/Q$ for each additional kilogram (or unit) of payload. In addition, distances d_{ij} between two locations i and j can be set as an adjusted distance that already reflects the effect of road gradient.

Furthermore, in our case speeds are considered to be time-dependent, which means they vary according to the departure times and periods of the day. Also, to account for the effect of fluctuating speeds during congestion periods (with stop-and-go traffic conditions), we use the RTOT ratio presented by Turkensteen (2017), which is the ratio between fuel consumption under fluctuating speeds and fuel consumption under fixed speeds. Essentially, it represents the degree of underestimation from fixed speeds computations (Turkensteen, 2017). These ratios were calculated modeling realistic driving conditions using the CMEM fuel consumption model and many driving cycles, which contain a sequence of driving speeds over time. For example, if we take the case of the New York City (NYC) driving cycle, the RTOT ratio of 1.34 for a 7-ton vehicle indicate that its fuel consumption with fluctuating speeds in NYC is 34% higher than in the case of a fixed average speed of 11.3 km/h.

However time-dependent, speeds are not optimized for each leg of the routes in our problem, on the contrary to the well-established PRP approach (Bektas and Laporte, 2011). The PRP was presented for road freight transportation, meaning trips have several long legs traveled in roadways, where trained drivers can accurately determine driving speeds (see, e.g., the instances generated for the PRP are based on cities from the UK).

In contrast to road transportation, when distributing in large urban centers as in our case, trucks are driven in much smaller speeds when comparing to rural and long-distance transportation. This happens because of congestion, narrow streets and several stop-and-go traffic conditions (e.g., semaphores and crossings), among other factors. Given these characteristics, it might not be very useful to optimize speeds in dense urban environments, as drivers are not likely to drive under the optimized speeds, as suggested by Srivatsa Srinivas and Gajanand (2017). In other words, although speeds are not optimized in our case, they are imposed by the traffic conditions (i.e., with time-dependent and fluctuating speeds), mainly due to congestion during peak hours.

Thus, these characteristics make our problem a Practical Pollution-Routing Problem with Time-Dependent Speeds, hereafter simply referred to as PPRP-TD.

The time horizon for scheduling can be set as wanted, such as business hours, 24h to encompass a whole day (e.g., the case of off-hour deliveries), or according to specific depot's open hours. The geocoded location and elevation of stops, as well as the network distances and mean travel times (i.e., not the time-dependent ones) between all vertices can be freely retrieved using Google Maps APIs (Google, 2017).

Several time-dependent speed ratios can be found in Figliozzi (2012), as we further detail in Section 6. Although not mandatory (e.g., users can use ours or Figliozzi's speed ratios values), the modeling of time-dependent travel times and speeds of a specific region of interest can also be done (e.g., using Google Maps APIs). In this way, it is able to obtain speed ratios' that more accurately represent traffic and congestion conditions of the specific region, similar to our approach to obtain the RMSP speed ratio profile.

In this way, it is worth noting that our PPRP-TD is *"Practical"* regarding how many parameters and user inputs are needed when compared to other approaches (e.g., Bektas and Laporte, 2011; Franceschetti et al., 2013). In Section 6 we present further details on how these parameters and variables can be easily retrieved, such as FCR functions for different classes of vehicles and speed ratio profiles.

4 Mathematical Formulation

The Practical Pollution-Routing Problem with Time-Dependent speeds (PPRP-TD) is defined on a directed graph $G = (V, A)$, where vertices 0 and $n + 1$ represent the depot and are referred as *"source"* and *"sink"*, respectively. Let $N = V\backslash\{0, n + 1\} = \{1,2, \dots, n\}$ be the set of customer vertices. All feasible vehicle routes are represented by source-to-sink elementary paths in G. To simplify notation, null demands and service times are associated to the depot vertices 0 and $n + 1$ (i.e., $q_0 = q_{n+1} = s_0 = s_{n+1} = 0$). Moreover, to represent the scheduling time horizon of the problem (e.g., the depot open hours), a time window is associated with both depot vertices (i.e., $[\alpha_0, \beta_0] = [\alpha_{n+1}, \beta_{n+1}]$), where α_0 and β_0 are the earliest possible departure time from and the latest possible return to the depot, respectively.

Furthermore, it is also given a set $K = \{1,2, \dots, k\}$ of vehicles, and a set $M = \{1,2, \dots, m\}$ of time intervals that represents speeds and travel times variations (e.g., congestion) during the time horizon. Each time interval m is defined by a speed ratio r^m, a beginning time b^m and an ending time e^m.

Note that when vehicles are allowed to stay at the depot, the arc $(0, n + 1)$ must be added to A, considering $d_{0,n+1} = t_{0,n+1} = 0$. Also, some arcs $(i, j) \in A$ can be omitted due to temporal considerations, if $\alpha_i + s_i + t_{ij}/r^m > \beta_j$ (in the case of customer time windows), or capacity limitations, if $q_i + q_j > Q$ (Toth and Vigo, 2014). Being $S \subseteq V$ an arbitrary subset of vertices, we use Toth and Vigo (2014)'s notation to represent in-arcs and out-arcs in directed graphs $G = (V, A)$ as follow: $\delta^-(S) = \{(i, j) \in A | i \notin S, j \in S\}$ and $\delta^+(S) = \{(i, j) \in A | i \in S, j \notin S\}$, respectively. For singleton sets $S = \{i\}$ it is used the notation $\delta^-(i)$ to represent the subset of arcs that arrive at vertex i and $\delta^+(i)$ for those that leave vertex i.

Finally, ρ_0 represents the vehicle no-load FCR and its counterpart ρ^* represents the full-loaded vehicle FCR, while R_0 is the RTOT ratio for an empty vehicle and R_{add} per additional ton, as mentioned in the previous section. Table 8 summarizes the used notation and some of the adopted values.

In this way, the PPRP-TD can be modeled by introducing four sets of decision variables:

(i) x_{ijk}^{m} binary variable that equals 1 if vehicle k travels from vertex i to vertex j in time interval m ($(i,j) \in A$; $k \in K$; $m \in M$);

(ii) $y_{ijk}^{m} \geq 0$ continuous variable that represents the payload (weight of cargo) carried by vehicle k when traveling from vertex i to vertex j in time interval m ($(i,j) \in A$; $k \in K$; $m \in M$);

(iii) $T_{ik} \geq 0$ continuous variable that specifies the start of service time at vertex i when serviced by vehicle k ($i \in V$; $k \in K$);

(iv) $D_{ik} \geq 0$ continuous variable that specifies the departure time from vertex i when serviced by vehicle k ($i \in V \backslash \{n+1\}$; $k \in K$).

Table 8 – PPRP-TD notation.

Notation	Information	Adopted values
n	Total number of customers.	–
k	Total number of vehicles.	–
m	Total number of time intervals.	–
V	Set of vertices.	$V = \{0,1,2,\dots,n+1\}$
N	Set of customers.	$N = V \backslash \{0, n+1\} = \{1,2,\dots,n\}$
M	Set of time intervals.	$M = \{1,2,\dots,m\}$
A	Set of arcs.	$A \subset \{(i,j) \mid i,j \in V\}$
K	Set of vehicles.	$K = \{1,2,\dots,k\}$
$\delta^{-}(i)$	Subset of arcs that arrive at vertex i.	$\delta^{-}(i) \subset A$
$\delta^{+}(i)$	Subset of arcs that leave vertex i.	$\delta^{+}(i) \subset A$
q_i	Demand of customer i.	(kg)
α_i	Start of time window associated with vertex i.	(h)
β_i	End of time window associated with vertex i.	(h)
s_i	Service time for customer i.	(h)
d_{ij}	Network distance from vertex i to j.	(km)
t_{ij}	Baseline travel time from vertex i to j (i.e., for $r^m = 1$).	(h)
Q	Capacity of the vehicles.	(kg)
ρ_0	Fuel consumption rate of the vehicles when empty loaded.	(L/km)
ρ^*	Fuel consumption rate of the vehicles when fully loaded.	(L/km)
R_0	RTOT ratio of an empty truck.	–
R_{add}	RTOT ratio per additional ton.	–
r^m	Speed ratio in time interval m.	–
b^m	Beginning of time interval m.	(h)
e^m	End of time interval m.	(h)
W	Maximum allowed working hours.	(h)
L_{ij}, L_1 and L_2	Large numbers.	–

We obtain the following Mixed-Integer Linear Programming (MILP) model:

$$\text{minimize} \sum_{m\epsilon M} \sum_{k\in K} \sum_{(i,j)\in A} d_{ij} \frac{1}{r^m} \left(R_0\, \rho_0\, x_{ijk}^m + R_{add}\, \frac{\rho^* - \rho_0}{Q} y_{ijk}^m \right) \tag{9}$$

Subject to:

$$\sum_{m\epsilon M} \sum_{k\in K} \sum_{j\in\delta^+(i)} x_{ijk}^m = 1 \qquad \forall\, i \in N \tag{10}$$

$$\sum_{m\epsilon M} \sum_{j\in\delta^+(0)} x_{0jk}^m = 1 \qquad \forall\, k \in K \tag{11}$$

$$\sum_{m\epsilon M} \left(\sum_{i\in\delta^-(j)} x_{ijk}^m - \sum_{i\in\delta^+(j)} x_{jik}^m \right) = 0 \qquad \forall\, k \in K, j \in N \tag{12}$$

$$\sum_{m\epsilon M} \sum_{i\in\delta^-(n+1)} x_{i,n+1,k}^m = 1 \qquad \forall\, k \in K \tag{13}$$

$$y_{ijk}^m \le Q x_{ijk}^m \qquad \forall\, k \in K, (i,j) \in A, m \in M \tag{14}$$

$$\sum_{i\in\delta^-(j)} y_{ijk}^m - \sum_{i\in\delta^+(j)} y_{ijk}^m = \sum_{i\in\delta^-(j)} q_j x_{ijk}^m \qquad \forall\, k \in K, (i,j) \in A, m \in M \tag{15}$$

$$T_{ik} + s_i + \frac{t_{ij}}{r^m} x_{ijk}^m - T_{jk} \le \left(1 - x_{ijk}^m\right) L_{ij} \qquad \forall\, k \in K, (i,j) \in A, m \in M \tag{16}$$

$$\alpha_0 \le T_{ik} \le \beta_0 \qquad \forall\, k \in K, i \in V \tag{17}$$

$$T_{jk} \ge D_{ik} + \frac{t_{ij}}{r^m} x_{ijk}^m + \left(x_{ijk}^m - 1\right) L_1 \qquad \forall\, k \in K, (i,j) \in A, m \in M \tag{18}$$

$$D_{ik} \le e^m + \left(1 - x_{ijk}^m\right) L_2 \qquad \forall\, k \in K, (i,j) \in A, m \in M \tag{19}$$

$$D_{ik} \ge \sum_{m\epsilon M} b^m x_{ijk}^m \qquad \forall\, k \in K, (i,j) \in A \tag{20}$$

$$D_{ik} \ge T_{ik} + s_i \qquad \forall\, k \in K, i \in V \tag{21}$$

$$x_{ijk}^m \in \{0,1\} \qquad \forall\, k \in K, (i,j) \in A, m \in M \tag{22}$$

$$D_{ik}, T_{ik} \in \mathbb{R}^+ \qquad \forall\, k \in K, i \in V \tag{23}$$

$$y_{ijk} \in \mathbb{R}^+ \qquad \forall\, k \in K, (i,j) \in A \tag{24}$$

The objective function (9) minimizes the fuel consumption of all vehicles. Constraints (10) ensure that each customer is assigned to exactly one route. Constraints (11–13) define a source-to-sink path in G for each vehicle k. Next, constraints (14) limit the maximum payload of any vehicle to Q, and force y_{ijk} to be zero when $x_{ijk}^{m} = 0$. Constraints (15) models the flow of cargo for all customers, requiring that the unloaded weight at customer j is equal to q_j, which also work as sub-tour elimination constraints (Suzuki, 2016; Xiao et al., 2012). Constraints (16–17) guarantee schedule feasibility within the defined time horizon. Constraints (18) compute the departure time at node i, and link the departure time and the start of service time variables, allowing vehicles to wait at a customer vertex if necessary. Temporal constraints (19–20) ensure that the departure time from vertex i sets the proper travel time between vertices i and j, based on the time interval m. Additionally, constraints (21) makes sure that vehicles only depart from a vertex after the moment of its arrival plus the corresponding service time. Finally, constraints (22) define the arc-flow variables to be binary and constraints (23) and (24) sets the positive continuous domains for all time-related variables and payload variables, respectively.

It should be noted that constraints (12) are not mandatory and can be omitted because the flow of vehicles through intermediate nodes in source-to-sink graphs are already modeled by temporal constraints (16), (18) and (21). In addition, it is possible to set customer time windows by generalizing (17) in the form of (25). Moreover, by using constraints (26) it is possible to guarantee that the total time spent by each vehicle (i.e., since its departure from the depot to its arrival back at it again) does not exceed the drivers' maximum allowed working hours. Also, constraints (16) and (18) could be rewritten without the variable x_{ijk}^{m} from the terms $\frac{t_{ij}}{r^m} x_{ijk}^{m}$, as in (27) and (28), respectively.

$$\alpha_i \leq T_{ik} \leq \beta_i \qquad \forall\, k \in K, i \in V \qquad (25)$$

$$T_{n+1,k} - D_{0k} \leq W \qquad \forall\, k \in K \qquad (26)$$

$$T_{ik} + s_i + \frac{t_{ij}}{r^m} - T_{jk} \leq \left(1 - x_{ijk}^{m}\right) L_{ij} \qquad \forall\, k \in K, (i,j,m) \in A \qquad (27)$$

$$T_{jk} \geq D_{ik} + \frac{t_{ij}}{r^m} + \left(x_{ijk}^{m} - 1\right) L_1 \qquad \forall\, k \in K, (i,j,m) \in A \qquad (28)$$

5 Solution Method

To solve the PPRP-TD, we propose a two-step solution method, similar to the approach by Xiao and Konak (2017). The first step is based on a series of adaptations and enhancements to the Savings Algorithm (Clarke and Wright, 1964) to solve solely the PPRP. The second step consists of extensive time-dependent scheduling of all routes yielded by the first step, which provides a final solution to the PPRP-TD.

Firstly, the savings algorithm is modified to account for the PPRP objective function, which is based on the fuel consumption rate (given by Expression 29; Suzuki, 2016) instead of traveled distances or costs used in the original algorithm. We also propose an extended version of the method that uses the concept of enhanced merging and savings (Stanojević et al., 2013) and updates the savings list at every iteration.

$$\text{minimize} \sum_{m \epsilon M} \sum_{k \in K} \sum_{(i,j) \in A} d_{ij} \rho_{ijk} x_{ijk}^{m} = \sum_{m \epsilon M} \sum_{k \in K} \sum_{(i,j) \in A} d_{ij} \left(\rho_0 x_{ijk}^{m} + \frac{\rho^* - \rho_0}{Q} y_{ijk}^{m} \right) \qquad (29)$$

To further improve the method's efficiency, we also combine randomization with a multi-start strategy to explore different neighborhoods, in a similar way to Stanojević et al. (2013). Essentially, this approach matches the first principle of the Greedy Randomized Adaptive Search Procedure (GRASP) metaheuristic, which is to build solutions using a randomized greedy algorithm (Feo and Resende, 1995, 1989; Labadie et al., 2016). The second GRASP principle is to apply local search procedures to find improvements to these solutions, which is done using a set of neighborhood operators in our case. Thus, our PPRP solution method is a GRASP FCR-based extended savings algorithm (GRASP FESA) that is able to obtain high-quality solutions efficiently.

In sequence to the PPRP solution, extensive time-dependent scheduling of all routes is performed to finally solve the PPRP-TD.

5.1 FCR-based Savings Algorithm (FSA)

The savings algorithm is a widely-used constructive heuristic to solve capacitated vehicle routing problems (CVPR), initially proposed by Clarke and Wright (1964)'s seminal paper.

The method first assigns each $n \in N$ customer to be served by a different vehicle, thus creating n elementary routes. Then, it is computed a list of savings that evaluates every possible merger (concatenation) of two routes if it does not lead to a violation of capacity (i.e., if the total load can fit in one vehicle) (Labadie et al., 2016). The evaluation of a given saving s_{ij} is done using Expression (30), in which i and j represent two elementary routes to be merged and d_{ij} is the distance (or cost) between customers i and j. The savings list is sorted in non-increasing order and, at each iteration, the merger with the largest positive saving (smallest negative cost) is executed. The algorithm stops when a single route is obtained or the merging of any two remaining routes exceeds the vehicle capacity.

$$s_{ij} = d_{i0} + d_{0j} - d_{ij} \tag{30}$$

Adaptations and enhanced variations of the savings algorithm, as well as its combination with metaheuristics, have been proposed for several different vehicle routing problems. A simple variant can be found in Segerstedt (2014), while Stanojević et al. (2013) present an extended version using enhanced mergers and a series of other improvements. Erdoğan and Miller-Hooks (2012), for instance, solved the Green Vehicle Routing Problem (GVRP) using a Modified Clarke and Wright Savings heuristic, alongside other techniques. The combination with metaheuristics includes Çatay (2010), who proposes a saving-based ant algorithm to solve the Vehicle Routing Problem with Simultaneous Pickup and Delivery (VRPPD), and Anbuudayasankar et al. (2012), which combines modified savings heuristics and genetic algorithm to tackle the Bi-objective Vehicle routing problem with Forced Backhauls (BVFB).

In our case, the first modification to the original savings algorithm is the inclusion of the fuel consumption rate to the savings calculation. The fuel saving S_{IJ} given by merging any two routes I and J is calculated using Expression (31), in which Q_J is the total payload carried by route J when leaving the depot; I_0 and I_{last} refer to the first and last customers of route I, respectively; J_0 and J_{last} are the first and last customers of route J, respectively; and D_I^* is the total distance for route I minus its last leg (i.e., the return to the depot $d_{I_{last},0}$), which is

calculated using Expression 32. Also, it is important to note that for any given pair of routes $\{I, J\}$, there are different possibilities of concatenations. According to Labadie et al., (2016), in the case of undirected networks with symmetric distances (or costs), four merging concatenations are possible, because each route can be inverted or not (see, e.g., Figure 3, from Labadie et al., 2016). However, when dealing with directed networks with asymmetric costs, eight mergers must be evaluated (Labadie et al., 2016). In this way, the mergers for a pair of routes $\{I, J\}$ in an asymmetric network is represented by the set S_{IJ} (Expression 33), in which $s_{IJ}^{*} = \max(s \in S_{IJ})$ represent the best saving from the set (i.e., the one with maximum value).

In addition, because the fuel consumption is directly related to the payload for each arc, the savings list must be updated to correct all savings involving the routes from the current merging iteration, which is dynamically done at each iteration. The pseudocode for the FSA heuristic is presented in Algorithm 1. Essentially, the FSA is the original savings algorithm (Clarke and Wright, 1964) with a different saving computation (Expression 31) and that updates the savings list at every iteration.

$$s_{IJ} = \rho_0\left(d_{I_{last},0} + d_{0,J_0} - d_{I_{last},J_0}\right) + Q_J \frac{\rho^{*} - \rho_0}{Q}\left(d_{0,J_0} - d_{I_{last},J_0} - D_I^{*}\right) \tag{31}$$

$$D_I^{*} = \sum_{\substack{(p,q)\in I,\\ q\neq 0}} d_{pq} \tag{32}$$

$$S_{IJ} = \{s_{IJ}; s_{I\_inv,J}; s_{I,J\_inv}; s_{I\_inv,J\_inv}; s_{JI}; s_{J\_inv,I}; s_{J,I\_inv}; s_{J\_inv,I\_inv}\} \tag{33}$$

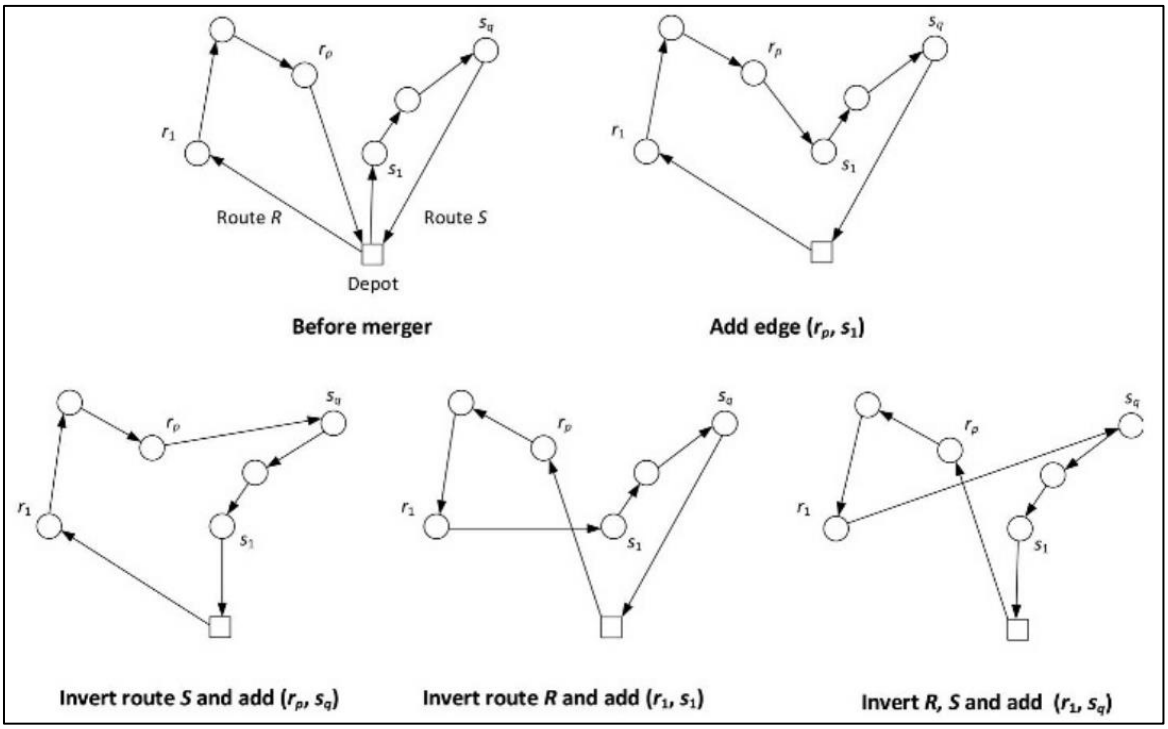

Figure 3 – Possible mergers in the savings algorithm (for undirected networks) (Labadie et al., 2016).

Algorithm 1 – FSA for the PPRP

Start: FSA for the PPRP.
1: Generate n elementary routes $I = (0, i, 0)$ and add them to the set R of routes;
2: **For** $(each\ route\ I \in R)$ **do**:
3: **For** $(each\ route\ J \in R, I \neq J, if\ Q_I + Q_J < Q)$ **do**:
4: Calculate all savings $s \in S_{IJ}$;
5: **If** $(any\ s \in S_{IJ} > 0)$ **then**
6: Add $s_{IJ}^* = \max(s \in S_{IJ})$ to the list L of savings;
7: **End − if**;
8: **Next** $route\ J$.
9: **Next** $route\ I$.
10: Sort the list L of savings in descending order of values;
11: **For** $(each\ saving\ s_{IJ}^* \in L)$ **do**:
12: **If** $(L = \emptyset)$ **then**
13: **STOP**;
14: **Else**
15: Merge routes I and J into route I;
16: Remove saving s_{IJ}^* from the list L;
17: Update the list L of savings and the list R of routes;
18: **End − if**;
19: **Next** $saving\ s_{IJ}^*$;
End: FSA for the PPRP.

5.2 FCR-based Extended Savings Algorithm (FESA)

Stanojević et al. (2013) presented the Extended Savings Algorithm (ESA), proposing a new way of merging routes and a corresponding formula for calculating the savings (called by the authors as enhanced savings). In our work, to further improve the efficiency of the FSA heuristic, enhanced savings are also evaluated during the savings list calculation, using the aforementioned authors' approach.

For a given pair of routes $M = (0, \dots, i, j, \dots, 0)$ and $N = (0, k, \dots, l, 0)$ without common vertices in exception for the depot, the enhanced merging of routes M and N yields a new route $P = (0, \dots, i, k, \dots, l, j, \dots, 0)$. The merging replaces the arc $a = (i, j) \in M$ by the path $(i, k, \dots, l, j)$ that is obtained from the route N, in which arcs $(0, k) \in N$ and $(l, 0) \in N$ are replaced by new arcs (i, k) and (l, j), respectively (Stanojević et al., 2013). This merging allows us to obtain the enhanced saving $S_{N,a}$, computed using Expression (34). In this way, an enhanced merging can also be seen as the insertion of route N in place of the arc $a \in M$.

$$S_{N,a} = d_{ij} + d_{0k} + d_{l0} - d_{ik} - d_{lj} \tag{34}$$

In the case of our FCR-based Extended Savings Algorithm (FESA) for the PPRP, the calculation of enhanced savings has also been modified to account for fuel consumption, not only corresponding distances. This is done using Expression (35) that calculates the enhanced saving S_{N,a_M} of merging route N within the arc $a \in M$. In Expression (35), $\breve{D}_M$ refers to the total distance for the path $(0, \dots, i)$ from route M and is calculated by Expression (36); D_N is the total distance from route N; Q_M^* is the cumulative load for customers j to M_{last} from route M; and Q_N is the total load carried by the route N when leaving the depot.

For both the ESA (Stanojević et al., 2013) and our FESA heuristic, when computing an enhanced saving, the procedure evaluates inserting a given example route in place of all the arcs from the other route (in exception of its first and last legs, as these are already evaluated by the concatenations of FSA). The algorithms then choose the best possible merger to be executed (i.e., the one that yields the highest positive saving).

$$\begin{aligned} S_{N,a_M} = {} & \rho_0\left(d_{ij} + d_{0k} + d_{l0} - d_{ik} - d_{lj}\right) + Q_N \frac{\rho^* - \rho_0}{Q}\left(d_{0k} - d_{ik} - \breve{D}_M\right) \\ & + Q_M^* \frac{\rho^* - \rho_0}{Q}\left(d_{ij} + d_{0k} + d_{l0} - d_{ik} - d_{lj} - D_N\right) \end{aligned} \tag{35}$$

$$\breve{D}_M = \sum_{\substack{(p,q)\in M, \\ p<i}} d_{pq} \tag{36}$$

$$Q_M^* = \sum_{p=j}^{M_{last}} q_p \tag{37}$$

In a similar way to the original savings algorithm and our FSA, in both the ESA and FESA heuristics four different mergers must be evaluated for any pair of routes, given that one or both routes can be inverted in the calculation. Also, in asymmetric networks, four additional evaluations are needed. Figure 4 presents an illustration of all possibilities of enhanced merging given any two routes. In Figure 4, the blue arcs represent the route M (or its inverse $M_{\_inv}$) and the red ones represent the route N (or $N_{\_inv}$). In addition, dotted lines refer to the arcs to be removed from the current routes M and N (or their respective inverses $M_{\_inv}$ and $N_{\_inv}$), while the black lines are the arcs to be added in order to construct the new route. In essence, the removed arcs (dotted) are the first and last legs from the route N, which is inserted in place of the arc $a_M \in M$ (or $a_N \in N$ in the case of the insertion of route M within N).

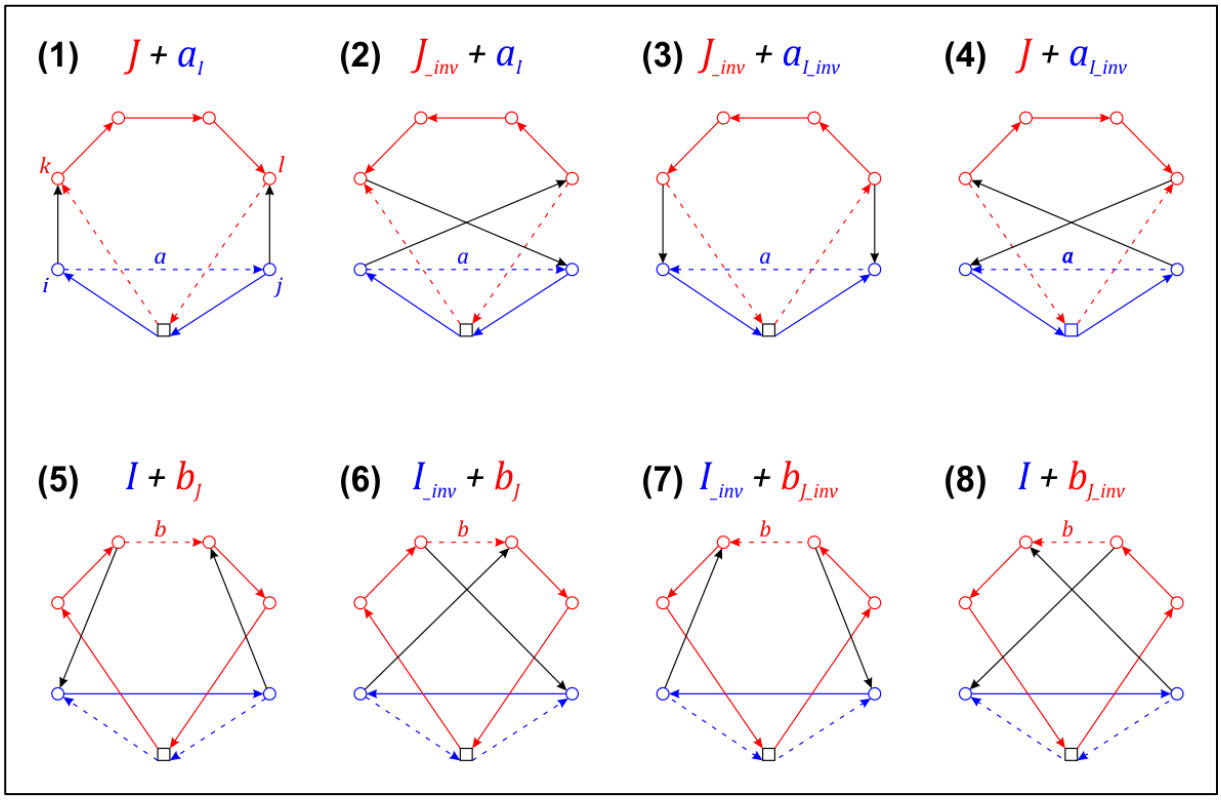

Figure 4 – Possibilities of enhanced merging.

5.3 Greedy Randomized Adaptive Search Procedure FESA (GRASP-FESA)

In a similar way to Stanojević et al. (2013), we introduce a randomization step to our algorithm combined with a multi-start strategy, with the aim to further explore the solution space using a greedy algorithm to obtain a more diverse set of solutions.

As previously mentioned, this approach matches the first principle of the Greedy Randomized Adaptive Search Procedure (GRASP) metaheuristic, which is to build a set of solutions using a randomized greedy algorithm (Feo and Resende, 1995, 1989; Labadie et al., 2016). At each iteration, on top of the incumbent randomized greedy solution, we apply local searches to improve it, which is the second GRASP principle and is performed using a set of three neighborhood operators.

In this way, our PPRP solution method is a GRASP FCR-based extended savings algorithm (GRASP-FESA). The pseudocode for our GRASP-FESA is presented in the form of Algorithm 2. Both the randomization step and local search procedures are detailed in the next subsections.

Algorithm 2 – GRASP-FESA for the PPRP

Start: GRASP-FESA for the PPRP.
Generate an initial solution x_0 using FESA;
$x_{best} \leftarrow x_0$;
$improvement\_flag \leftarrow 1$;
For $(i = 1\ to\ \tau\ restarts)$ **do**:
    Generate a random x solution using Randomized FESA;
    While $(improvement\_flag = 1)$ **do**
        $x' \leftarrow Node\_Relocate(x)$;
        $x'' \leftarrow Node\_Swap(x')$;
        $x''' \leftarrow 2\_\mathrm{Opt}(x'')$;
        If $\left(f(x''') < f(x)\right)$ **then**
            $improvement\_flag \leftarrow 1$;
        Else
            $improvement\_flag \leftarrow 0$;
        End − if;
        If $\left(f(x''') < f(x_{best})\right)$ **then**
            $x_{best} \leftarrow x'''$;
        End − if;
    End − while;
Next i.
Return (x_{best})
End: GRASP-FESA for the PPRP.

5.3.1 Randomization

Combining a multi-start strategy with randomization in our greedy FESA is an easy way to obtain a pool of good solutions and improve the method's efficiency, as pointed out by Stanojević et al (2013). Our Randomized FESA (R-FESA) uses a parameter τ that sets the number of restarts of the algorithm (i.e., number of seeds) and adds randomization when choosing a saving to be executed and merge two routes.

In this way, the randomness to diversify the search is introduced at each merger, which chooses a random saving from the best α savings from the savings list L, instead of always choosing only the best one to execute (i.e., the one with maximal value, like FESA). The chosen saving can be selected by using an uniform or any biased distribution (e.g., favoring bigger savings) (Stanojević et al., 2013). According to Feo and Resende (1995), this subset containing the best α savings (candidates) from L is called the Restricted Candidate List (RCL) and allows for different solutions to be obtained at each GRASP iteration, while not necessarily compromising the quality of our greedy algorithm.

5.3.2 Local Searches

In the improvement phase of our GRASP-FESA, a set of three classical neighborhood-based local search operators are applied to further improve the method's efficiency: $\mathcal{L}_1$, a Relocate move; $\mathcal{L}_2$, an Exchange move; and $\mathcal{L}_3$, a 2-opt move. At each GRASP iteration, after obtaining a random greedy solution, we firstly perform the Relocate move $\mathcal{L}_1$ on the solution. $\mathcal{L}_1$ iterates through all nodes from each route of the solution and checks if moving it from the current position to another position from the given route (or any other route) yields an improvement to the total fuel consumption of the solution. An example of relocating a node within a route is presented in Figure 5 – Relocate move., in which node i is moved from its current position to another position in the same route, by replacing arcs $(i-1, i)$ and $(i, i+1)$ by arc $(i-1, i+1)$ and arc $(j, j+1)$ by arcs (j, i) and $(i, j+1)$.

Following $\mathcal{L}_1$, the Exchange (or swap) move $\mathcal{L}_2$ is performed, also iterating through all nodes from the current solution. $\mathcal{L}_2$ aims to improve the total fuel consumption by exchanging the

position of two nodes within a route or swapping two nodes between two different routes, if the move yields to an improvement to the objective function. A relocate move within a route is illustrated in Figure 6; arcs $(i-1, i)$ and $(i, i+1)$ are replaced by arcs $(i-1, j)$ and $(j, i+1)$ respectively, while arcs $(j-1, j)$ and $(j, j+1)$ are replaced by arcs $(j-1, i)$ and $(i, j+1)$ respectively. The images depicted in Figures 5 and 6 represent only intra-route examples of the local search operators, but our implemented local searches also accounts for both intra-route and inter-routes moves.

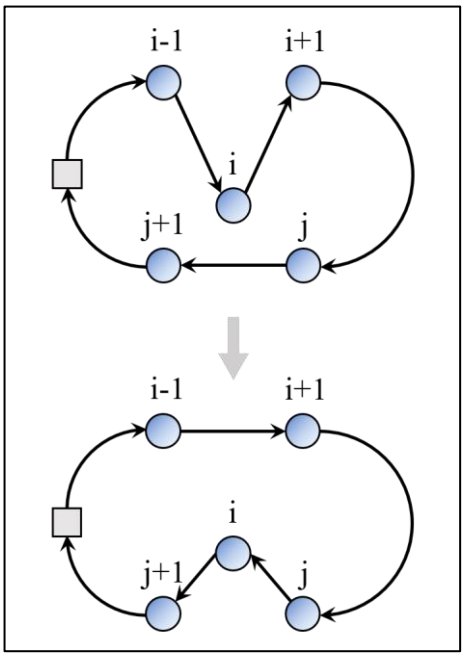

Figure 5 – Relocate move.

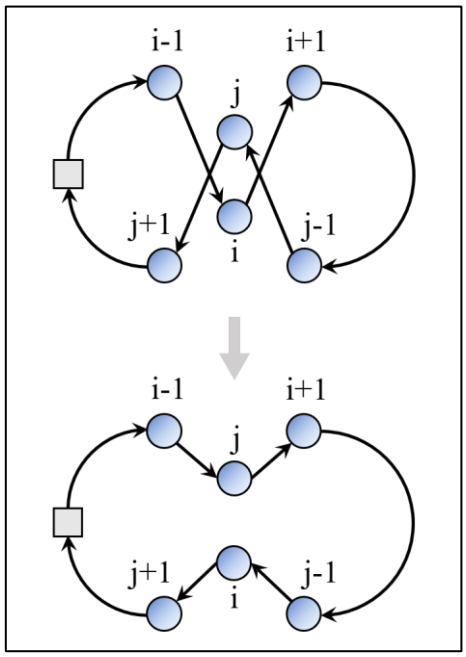

Figure 6 – Exchange move (swap).

Lastly, the 2-opt operator $\mathcal{L}_3$ is performed. Unlike $\mathcal{L}_1$ and $\mathcal{L}_2$ that iterates through all nodes, $\mathcal{L}_3$ iterates through all routes but verifying if arcs (instead of nodes) from given routes can be substituted by other arcs, seeking to remove crossings for example (e.g., Figure 7), thus leading to improved solutions. In the case of $\mathcal{L}_3$, an excerpt from the route should also be inverted (i.e., all arcs between the two replaced arcs) to obtain a valid feasible solution. In Figure 7, arcs $(i, i+1)$ and $(j, j+1)$ are replaced by arcs (i, j) and $(i+1, j+1)$ respectively, while arc $(i+1, j)$ is inverted into $(j, i+1)$.

Another characteristic of our GRASP-FESA is that the improvement phase containing operators $\mathcal{L}_1$, $\mathcal{L}_2$ and $\mathcal{L}_3$ are executed within a loop (Algorithm 2: lines 6 to 18) until no further improvement can be found because the performed moves from the current iteration might lead to new moving possibilities in following iterations.

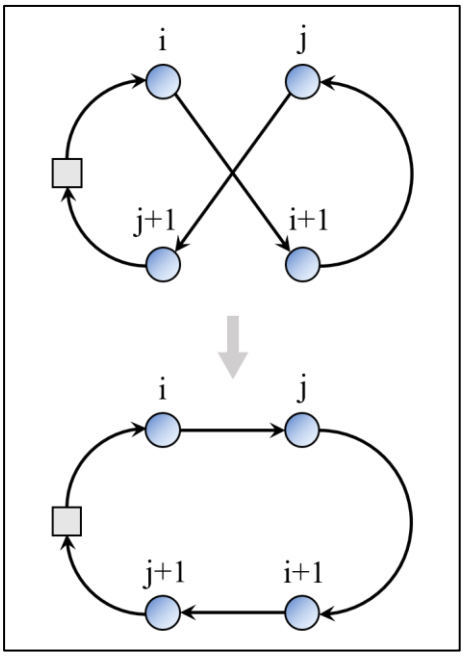

Figure 7 – 2-opt move.

5.4 Time-Dependent Scheduling

To finally solve the PPRP-TD, an extensive Time-Dependent Scheduling (TD Scheduling) is performed in the final best solution yielded by GRASP-FESA. Given a time-step ε expressed in the same unit of the time horizon (e.g., minutes), a schedule time-table for all routes is enumerated, discretizing the continuous time horizon domain into $\left\lfloor {\beta_0}/{\varepsilon} \right\rfloor$ parts, with vehicles departing $p \in \{1, \dots, \left\lfloor {\beta_0}/{\varepsilon} \right\rfloor\}$ times at every ε minutes from the beginning of the time horizon α_0 to its end at β_0 (i.e., $\alpha_0 < p * \varepsilon < \beta_0$). This enumeration of schedules allows us to choose the best solution among those discretized. Although it does not guarantee us to obtain optimal solutions, it still produces good solutions as solutions can be computed in $O(n)$ time and the time horizon can be discretized in more parts by using smaller ε values.

Also, being the PPRP-TD a time-dependent problem, it must respect the First-In-First-Out (FIFO) property. This property, also referred as the "non-passing" property (Kuo, 2010), ensures that for vehicles departing from a store i to a store j, an earlier departure time always produces an earlier arrival time, and vice versa. Kuo (2010) presents an example of a "passing" occurrence in Figure 8, for a given arc (i, j) with $d_{ij} = 60km$. As illustrated, a vehicle that

departs from a customer at 13:15 arrives to its destination before another vehicle that leaves the same customer at an earlier time (12:45), thus disrespecting the FIFO property.

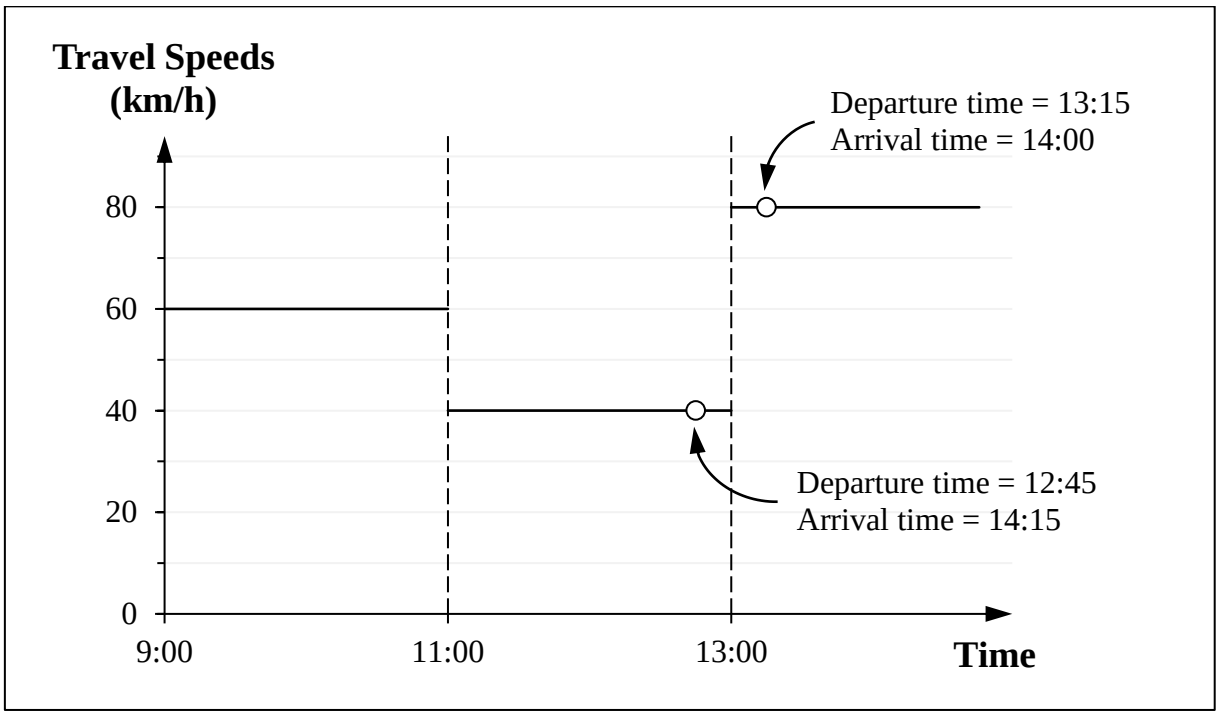

Figure 8 – Step functions of the travel speeds for a given arc of 60km, which illustrates an example of "passing" (adapted from Kuo, 2010).

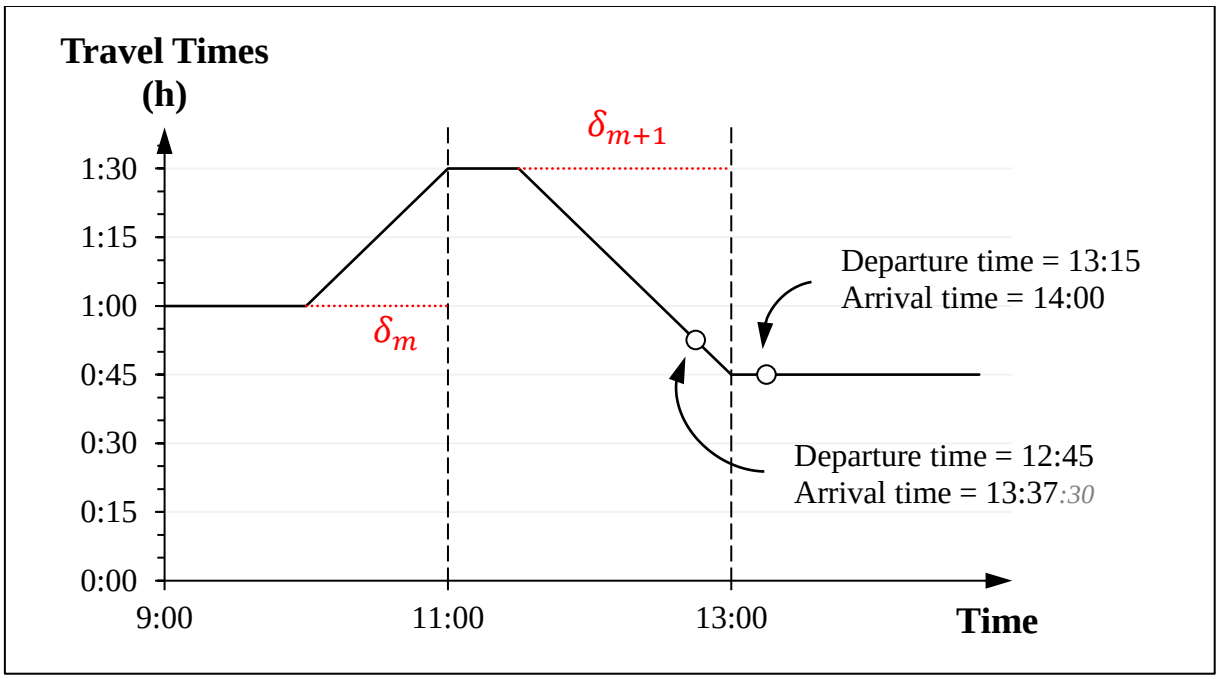

Figure 9 – Piecewise linear function of the travel times for a given arc of 60km, where "passing" does not occur.

In order to respect the FIFO property, we use the approach presented by Ichoua et al. (2003), in which travel speeds are given by step functions (the same as in our work) but travel times are modeled by a piecewise continuous function over time. The authors claim that although travel speeds also change continuously over time, the use of step functions to model travel speeds is more reasonable than for travel times. In this way, this can be illustrated in the same example presented in Figure 8. For the vehicle departing at 12:45, it would be like it travels the first 15 minutes of the trip with a 40km/h speed (i.e., traversing a length of 10km), and the remaining 50km of the trip the vehicle would be traveled with the higher speed of 80km/h after the moment when the change in travel speeds occurs.

To illustrate this approach and using the same scenario depicted by Kuo (2010), Figure 9 presents the aforementioned scenario using a continuous piecewise linear function to model travel times, in which the FIFO property is respected. The vehicle that departs at 12:45 would take only 52 minutes and 30 seconds to traverse the arc of 60km, instead of 1 hour and 30 minutes if travel time was calculated using solely the step travel speeds and departure time. The horizontal axes in Figures 8 and 9 are presented in the same scale and vehicles depart at the same time in both examples. However, only in the latter case (i.e., the piecewise function), a vehicle that leaves at a later time does not pass another one that left at an earlier time.

In this way, for any given arc (i, j) of length d_{ij} and average travel time t_{ij} (i.e., not time-dependent) and considering the current time interval m, the moment in which travel times begin to change occurs δ_m minutes (or other unit of time) before the boundary between the two consecutive time intervals m and $m + 1$ is reached. δ_m can be calculated using Expression (38).

$$\delta_m = \frac{2t_{ij}}{r^m + r^{m+1}} \tag{38}$$

If $\delta_m > e^m - b^m$ for any given arc and time interval, then δ_m assumes the value of $e^m - b^m$, thus increasing the pace of increase to travel times (or decrease if going from a faster time interval to a slower one). At this point, the travel time to traverse the arc is calculated by dividing d_{ij} and the travel speed for the current time interval m (i.e., t_{ij}/r^m). From this point up to until the beginning of the next time interval $m + 1$ (i.e., b^{m+1}), travel times are calculated using a linear function that starts at $\left(\frac{t_{ij}}{r^m}, e^m - \delta_m\right)$ and goes up to $\left(\frac{t_{ij}}{r^{m+1}}, b^{m+1}\right)$.

In this way, our previously mentioned time-table schedule enumeration uses this approach to schedule the departure and arrival times for all used vehicles and every visited node, thus respecting the FIFO property in the process. However, although our heuristic considers the

FIFO property, it is important to note that our mathematical formulation given by Expressions (9–28) do not respect it, as vehicles are allowed to wait before or after serving the customers.

6 Experiments

Given the Practical Pollution-Routing problem with time-dependent speeds presented, this section introduces new instances based on real retail distribution in São Paulo to test the efficiency of the proposed solution method. We also provide some insights on the influence of directions on travel speeds of the network of a megacity such as São Paulo, in which traffic congestion is of great influence, especially during peak hours.

Fuel Consumption Rate functions from the literature are also presented for different classes of trucks, which can be used as examples to practitioners. If more accuracy is needed, practitioners can model the fuel consumption of their fleet using one of the models presented in Section 2. We dive into this modeling by presenting how CMEM can be used with real GPS data to provide an FCR function for specific vehicles.

With all the data needed for the generation of instances, we performed an algorithm tuning to our GRASP-FESA, in order to the solution method yield the best solutions. Finally, we present the results for all of our computational experiments for both the PPRP and the PPRP-TD.

Further details on each of these subjects are presented in following subsections.

6.1 Generation of benchmark instances based on real data from São Paulo

To illustrate an application of the PPRP-TD in the distribution of a real-world company, we used empirical data from the Off-Hour Deliveries (OHD) pilot test in São Paulo (CISLOG, 2015; Cunha et al., 2016), conducted by the Center for Innovation in Logistics Systems (CISLog). Analysis from truck GPS data in São Paulo indicates that average speeds during night periods are about 40% greater than those measured during peak hours, both in the morning and the evening (CISLOG, 2015; Cunha et al., 2016). Moreover, OHD has a direct impact on urban freight emissions, as shown by Holguín-Veras et al. (2016), thus evidencing the effects on fuel consumption given by distributing in real-world scenarios with time-dependent speeds.

The company is a large retail company that operates in several states of Brazil. In São Paulo, there were almost 300 stores in the city in October 2014 (not considering the entire Metropolitan Region of São Paulo). Using Google APIs (i.e., Geocoding and Elevation), we obtained the

geographic locations (latitudes and longitudes coordinates) and elevations for a set of 258 stores plus the company's warehouse, totaling 259 vertices. Network distances (not symmetric) and travel times between every pair of vertices were also obtained using the Directions and Distance Matrix APIs (Google, 2017). It is worth noting that the Directions Matrix API also allows to set up a departure time in the query, meaning it is possible to obtain time-dependent travel times. Using this feature, we retrieved the time-dependent travel times between every pair of vertices for all hours of the day, as well as the mean travel times when no departure time was set. Then, assuming fixed network distances, we divided the distances of each arc by its corresponding mean travel time, to obtain a mean travel speed for every arc of the network. Furthermore, this division was carried out involving all distances and their corresponding time-dependent travel times for the 24 hours of the day, thus obtaining time-dependent travel speeds for every pair of vertices (i.e., all arcs) and every departure hour. Now, taking each set of time-dependent travel speeds (each set corresponding to an hour of the day) and dividing it by the set of mean travel speeds, we obtained sets of travel speed ratios for every hour of the day. It is worth mentioning that the final aggregated value for a given hour of the day was obtained by the average between the speed ratios of all arcs from the network.

The obtained speed ratios are illustrated by the continuous blue lines in Figures 10 and 11, representing the average of speed ratios for each hour of the day. The discrete red lines in both figures represent the speed ratios for the time periods aggregated by similar travel speed ratios, with values being the average of the corresponding time interval.

In our experiments, we chose to use a speed profile with $m = 5$, which matches the time dependent adaptation proposed by Figliozzi (2012) to the classical Solomon (1987)'s set of VRPTW instances. However, we also present one speed profile "*easier"* to be solved exactly ($m = 2$) that represents only day and night speeds (e.g., as in OHD). Figures 10 and 11 show the travel speed profile for $m = 5$ time intervals and $m = 2$, respectively. It is also shown the residual sum of squares (RSS) for each depicted scenario.

As the company operates its warehouse all day, the time horizon for the proposed instances is set to 24h, starting at 6 am. According to the real data, stores may have very wide time windows, meaning only day, night, or both. In other words, stores are divided into three groups: (i) those that receive deliveries during the day only (6 am to 9 pm); (ii) during the night only (9 pm to 6 am, e.g., stores in shopping malls); or (iii) during all day (24h stores). However, although we set these time windows for the newly proposed instances, they are not taken into account in our experiments, as our GRASP-FESA was not designed to consider time windows.

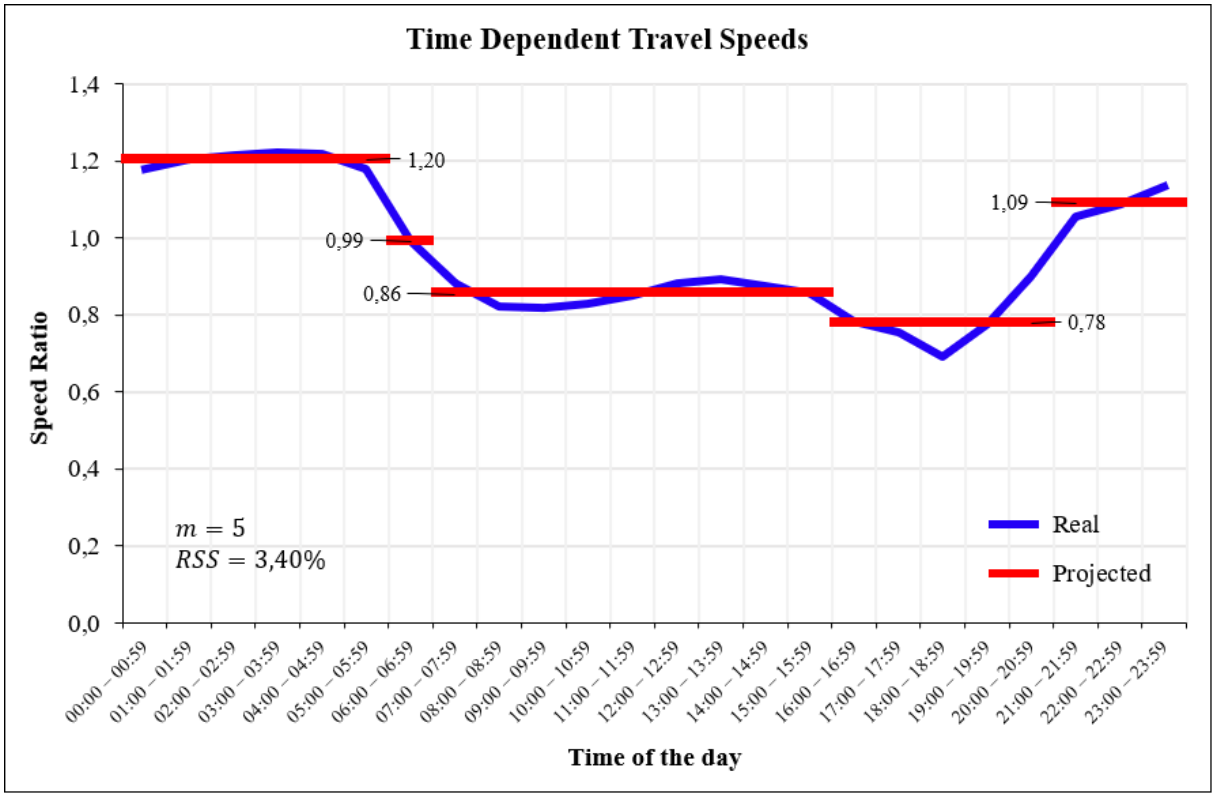

Figure 10 – Real and projected speed ratios (r^m) for different times of the day ($m = 5$).

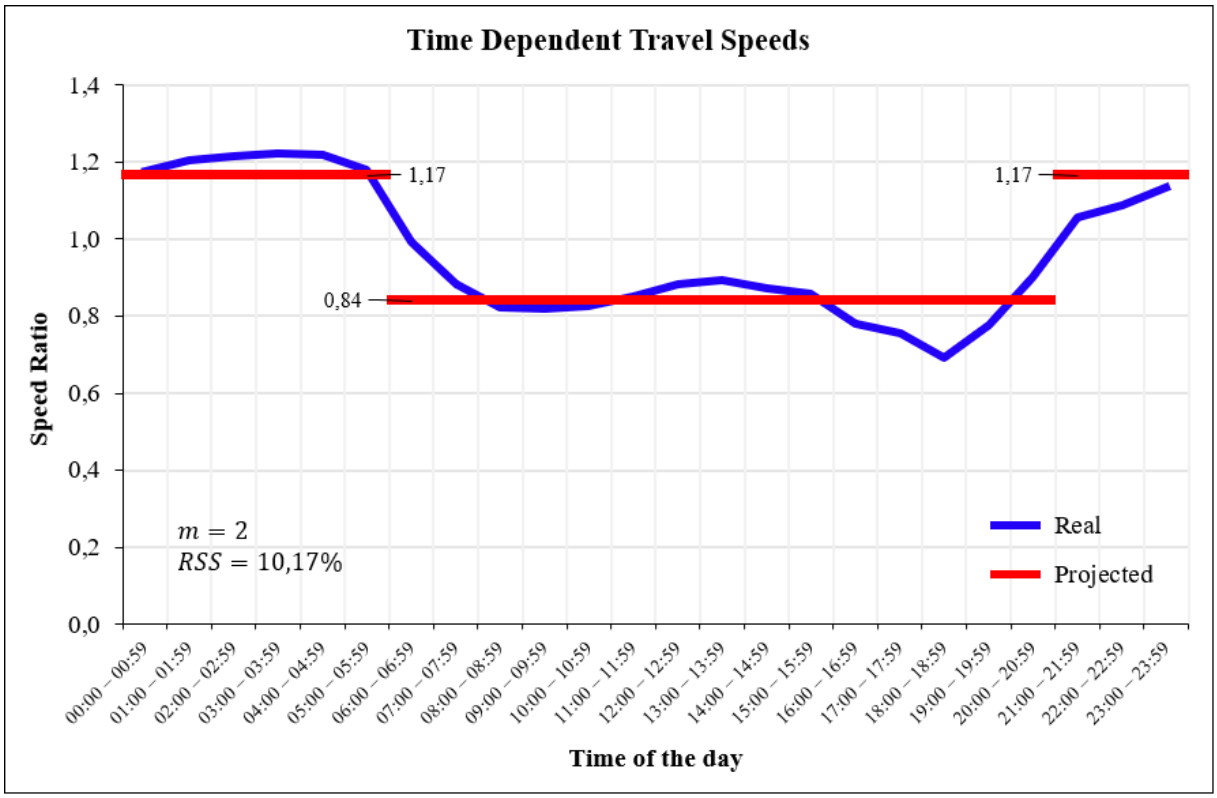

Figure 11 – Real and projected speed ratios (r^m) for different times of the day ($m = 2$).

In addition, based on empirical data (i.e., the demand values from 14 real delivery routes within roughly 70 customers during a three-month period of distribution), we could also analyze the company's drop size pattern (in number of boxes), as shown in the histogram in Figure 12. We approximated it to a log-normal distribution (mean 3.481 and standard deviation of 0.411) to randomly assign demands to all 258 customers.

Boxplots of the vehicle capacity usage for each of these routes is presented in Figure 13. By analyzing the figure, we could determine the vehicle capacity in terms of the number of loaded boxes. In this way, we set Q to 350 boxes.

Within the full set of 258 stores, we also created a smaller set of stores ($n = 58$) that contains stores located only in the west region of the city (namely SPZO, from Portuguese *"São Paulo Zona Oeste"*) to generate smaller instances. The SPZO set is illustrated in Figure 14. However, for the instances containing only these SPZO stores, we set Q to 250 boxes because of the small number of stores and to limit the number of stores that can be served by a single vehicle. It is also worth noting that in Figure 13, routes 4 and 11 have the smallest capacity usage because these delivery routes contained only few stores.

Using these previously defined parameters, stores' locations and demands, we generated five sets of small SPZO instances with different sizes (i.e., $n \in \{5,10,15,20,25\}$), to be solved exactly using the mathematical formulation proposed in Section 4. Each set contains five different instances with stores chosen randomly from the SPZO set of stores. In this way, there are 25 different small instances generated from the set of locations presented in Figure 14. We also generated large SP instances (i.e., $n \in \{50,100,200,250\}$) from the whole set of stores (all city), which can be solved using heuristics and approximation algorithms.

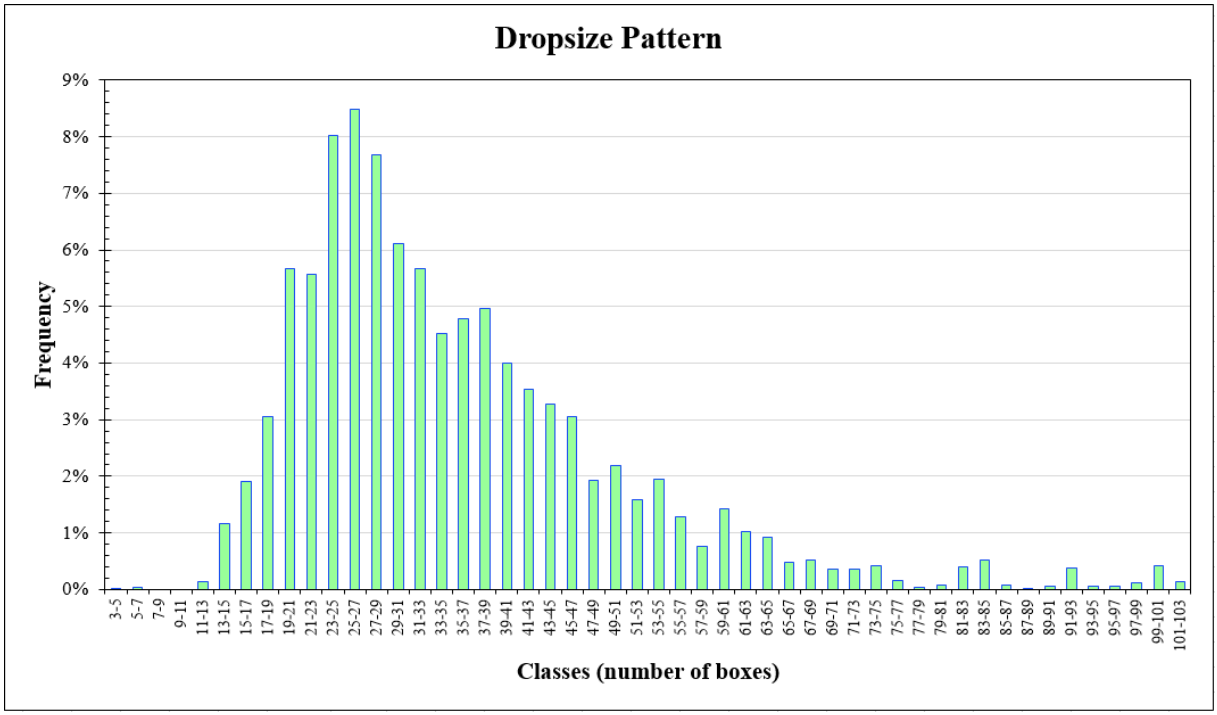

Figure 12 – Demand's histogram, resembling a log-normal distribution.

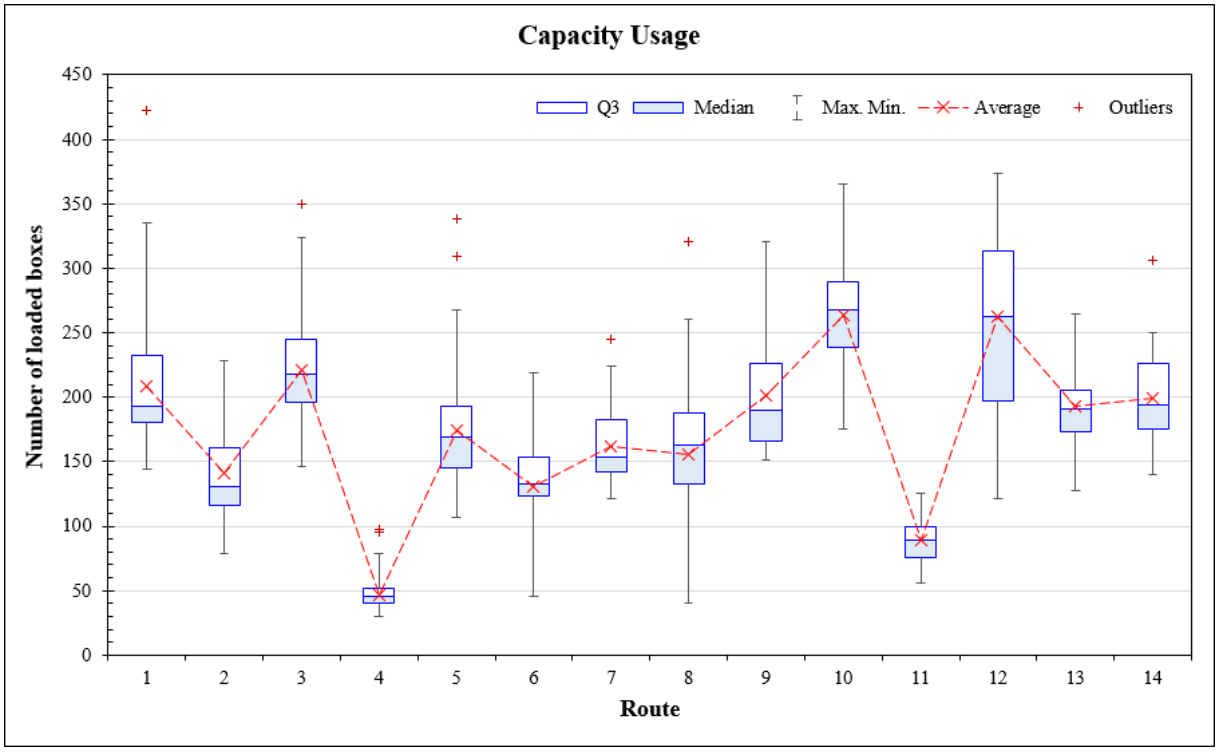

Figure 13 – Capacity usage of the vehicles.

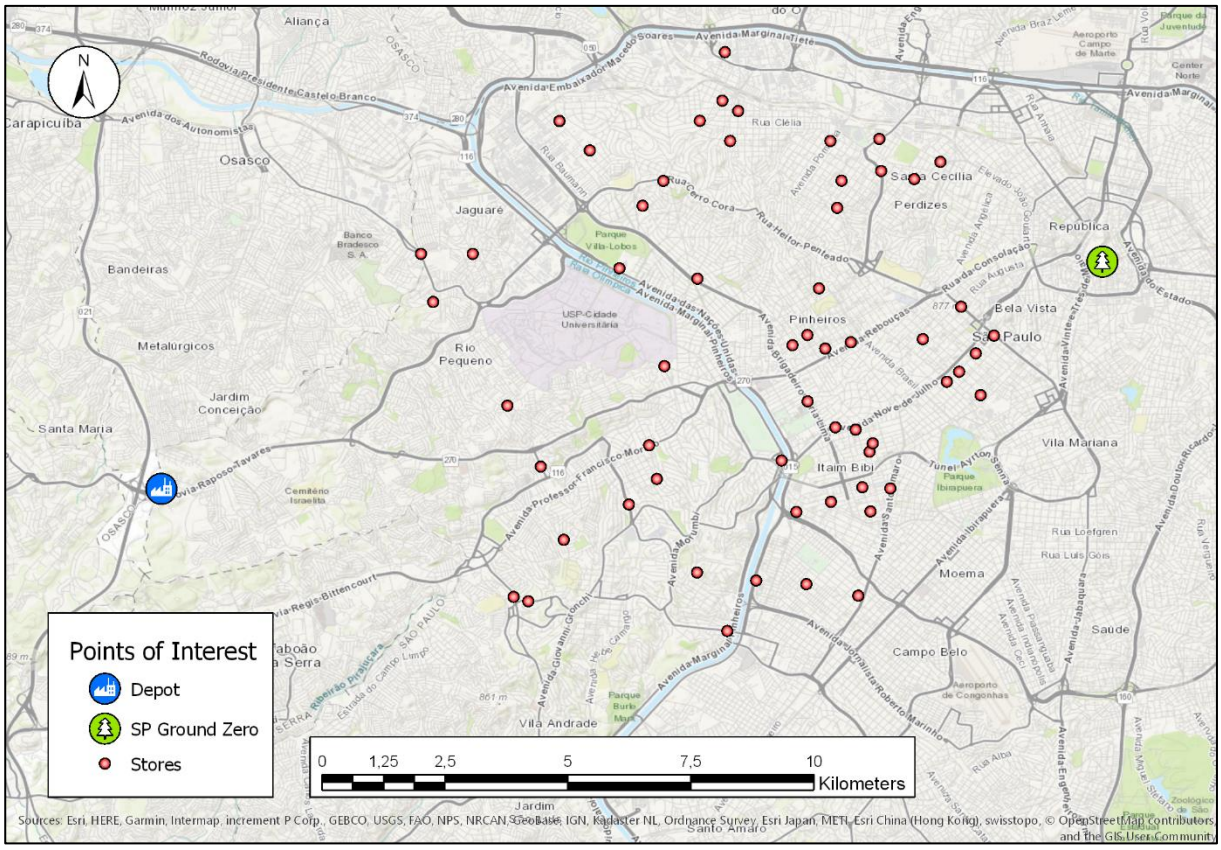

Figure 14 – Distribution of point locations used to generate the SPZO set of instances.

6.1.1 Directions matter?

When analyzing travel times and travel speeds of a megacity such as São Paulo, it is possible to clearly observe some characteristics of directed networks. In directed networks the matrix of distances, travel times and travel speeds are not symmetric, meaning that an arc (a, b) is not equivalent to arc (b, a). A clear example is given in the form of Figure 15, in which we compare the median travel speeds of all arcs going from the depot to all of the stores of the SPZO set and the other way around (i.e., going back from all customers to the depot). In Figure 15, the green line (depot to customers) represent the scenario where the flow is going directly into the city (radially from the outer ring directed to downtown, see Figure 14), which present the lowest travel speeds during the peak hours of the morning. In contrast, the opposite scenario depicted by the red line shows that lower speeds happens during the peak hours of the evening instead of during the morning.

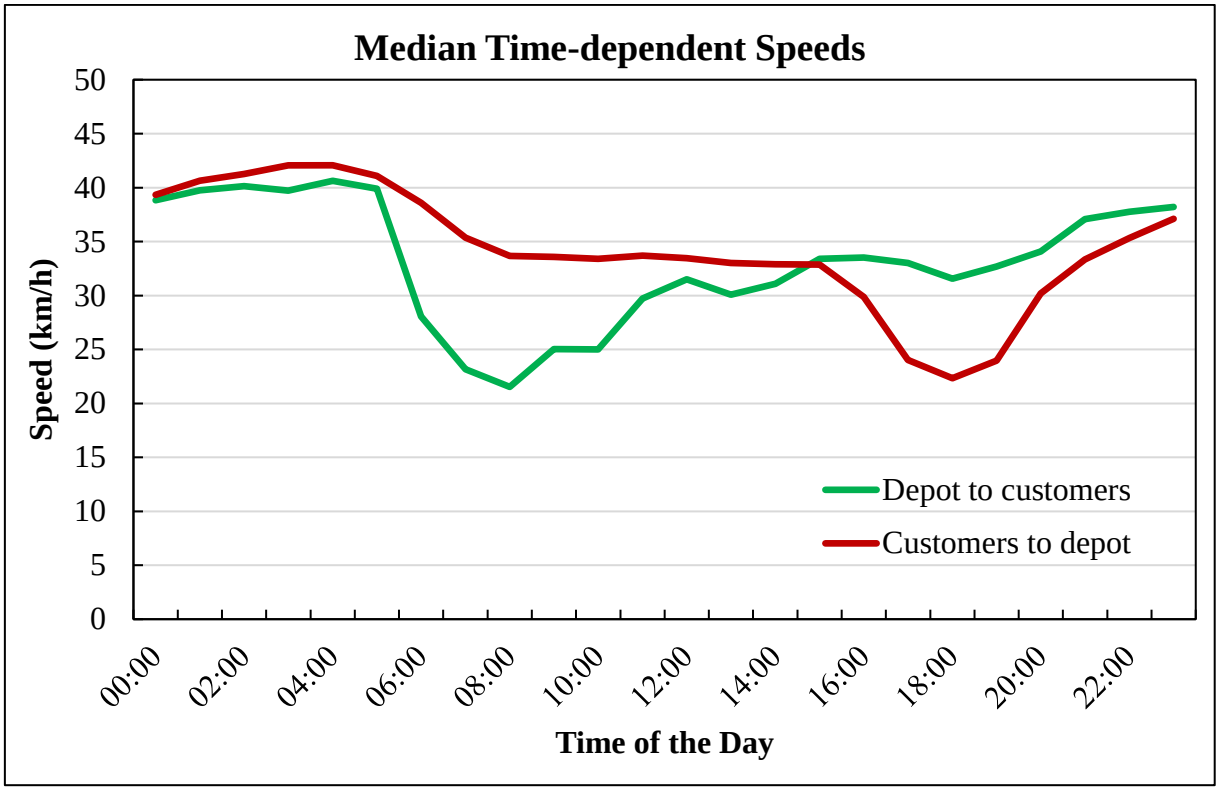

Figure 15 – Comparison of time-dependent speeds given different directions.

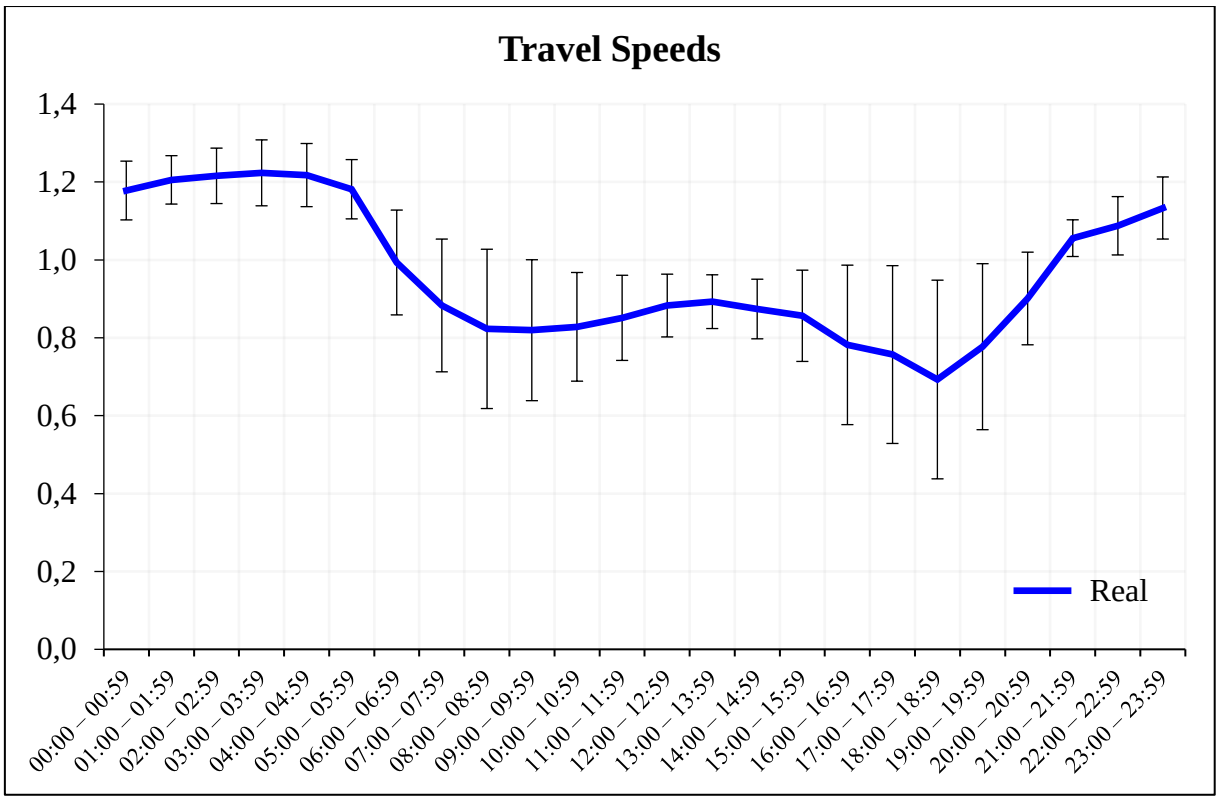

Figure 16 – Standard deviation of speeds.

This fact is evidenced in Figure 16, in which the aggregated values to form the curve represent the average of speed ratios between all arcs of the network for each hour of the day. By observing the standard deviation of each hour, it is possible to note that the higher variation occurs during the peak hours of the morning and the evening, while during the night, dawn and beginning of the afternoon there are lower variations. Though intuitive, this effect indicates that, when considering two arcs (a, b) and (b, a), traffic congestion during peak hours might occur only in one direction, while the other may present a free flow with no congestion.

In this way, as our analysis suggests, it is possible to say that directions indeed matter and should be considered (at least in large urban centers such as São Paulo, where congestion and time-dependent speeds is a relevant concern). This effect, however, is not taken into account in our experiments and we leave it for future research.

6.1.2 FCR functions

In line with the PPRP-TD, an FCR function is needed to calculate the fuel consumption in the experiments. Table 9 presents the fuel consumption rate functions for several different classes of vehicles found in the literature. These functions are also illustrated in Figure 17. Practitioners can use one of these functions or model a more accurate linear function based on GPS tracks of real distribution using its own fleet. We exemplify this modeling by using real GPS data to obtain an FCR function for a specific vehicle class: VUCs. The GPS data were gathered during another project from CISLog, in which the vehicle routing had a similar condition as in our case (e.g., vehicle class, weight transported, and number of stops).

Racelogic^{TM} GPS receivers model VBOX PRO with a 10 Hz sampling rate (Figure 18) were used. These GPS receivers were installed on trucks distributing in the urban environment of São Paulo and region. The available data had GPS tracks for 24 routes, being 18 of them distributing exclusively in urban scenarios, and the remaining 6 also delivering to nearby cities, thus referring to a different driving cycle (urban with some road transportation). These GPS tracks are summarized in Table 10.

Regarding the company's fleet, it owns a homogenous fleet of VW Delivery trucks, light-duty vehicles available in the size of VUCs that are allowed to drive within São Paulo's Zone of

Maximum Traffic Restrictions (ZMRC, from Portuguese *'Zona de Máxima Restrição de Circulação'*).

Finally, the modeling of fuel consumption was carried out using the comprehensive modal emission model (CMEM) (Barth et al., 2005), using Expressions (3–7) for each. By varying the total vehicle weight from 2000kg (empty vehicle) to 4500kg (fully loaded) for each of the 18 routes, we could obtain a range of FCR values, allowing an analysis of FCR variation per total vehicle weight. The fuel consumption variation for a VUC running in a large urban center is illustrated in Figure 19 in the form of boxplots. The aggregation of these values by their averages yielded in the FCR function $y = 0.0009 * x + 15.207$, also presented in Table 9. This FCR function and weight variation corresponds to a no-load FCR of $\rho_0 = 0.18807$, and to a full-load FCR of $\rho^* = 0.23307$.

In addition, the degree of underestimation in fuel consumption for an empty vehicle with a gross weight of 2.5 tons is $RTOT_0 = 1.21$, calculated using Turkensteen (2017)'s approach, considering the driving cycle of New York City (NYC) and the diesel density equal to 0.8347 kg/L, according to the conversion factor of British DEFRA (2010). Being a large urban center, the NYC driving cycle was chosen because of its similarities with São Paulo in terms of congestion and stop-and-go driving conditions. The NYC driving cycle present a low average speed of 11.3 km/h, as reported by Turkensteen, 2017. This value is close to the value of 14.5 km/h that we found for São Paulo when analyzing the GPS tracks of urban distribution in São Paulo and region (see Table 10). Also, according to Turkensteen (2017), the ratio for each additional ton of cargo running in an environment such as NYC is $RTOT_+ = 3{,}75$.

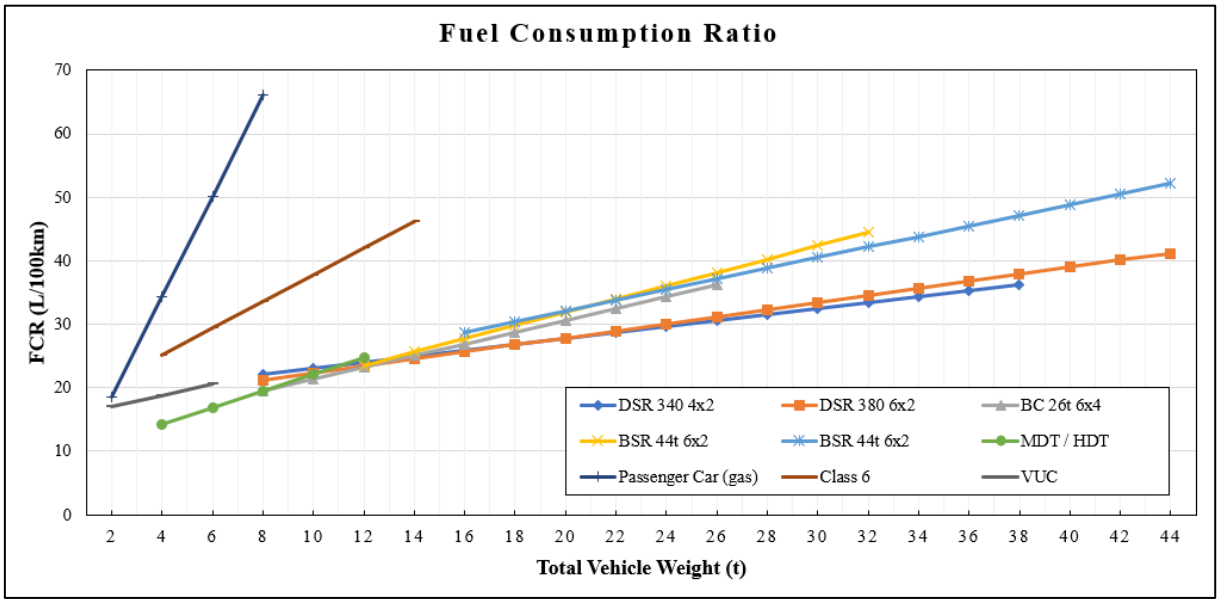

Figure 17 – Fuel consumption ratio for several different classes of trucks.

Table 9 – FCR functions for different classes of vehicles.

Source	Vehicle		GCW (t)	GVWR (t)	FCR Function
Rizet et al. (2012), Coyle (2007)	Articulated Trucks	DSR 340 4x2	38		$y = 0{,}4747 * x + 18{,}236$
		DSR 380 6x2	44		$y = 0{,}5563 * x + 16{,}723$
	Tippers	BC 26t 6x4		26	$y = 0{,}9266 * x + 12{,}078$
		BC 32t 8x4		32	$y = 1{,}045 * x + 10{,}984$
		BSR 44t 6x2		44	$y = 0{,}8391 * x + 15{,}3$
Wu et al. (2015)	Medium-Duty Trucks (MDT) and High-Duty Trucks (HDT)		6–36		$y = 1{,}315 * x + 8{,}938$
Xiao et al. (2012), Suzuki (2016)	Passenger Car (gas)		1.5		$y = 0{,}0000793 * x + 0{,}026$
Delorme et al. (2009)	Class 6 Trucks		12		$y = 0{,}0021 * x + 16{,}762$
Our GPS analysis	VUC		2–6		$y = 0{,}0009 * x + 15{,}207$

Figure 18 – GPS receiver equipped on the trucks (RACELOGIC, 2018).

Table 10 – Analyzed GPS routes.

Driving Cycle	Route	Total Distance (km)	Total Time (h)	Avg. Speed (km/h)	GPS Count
Urban	HR_1	83,23	9,28	8,97	33424
	HR_3	76,75	9,59	8,00	34527
	HR_4	88,00	4,42	19,93	15900
	HR_6	41,79	3,22	13,00	11578
	HR_27	70,54	5,12	13,78	18434
	HR_29	138,79	10,42	14,90	33426
	HR_30	88,70	5,83	28,03	20974
	HR_33	72,54	6,59	11,01	23725
	HR_34	75,27	7,91	9,52	28479
	HR_36	71,97	5,64	12,68	20261
	HR_38	73,98	6,55	11,30	23568
	HR_40	112,87	5,47	19,20	21161
	HR_44	114,27	7,32	15,59	26271
	HR_48	109,39	6,71	16,26	24098
	HR_51	96,00	7,12	13,44	25598
	HR_54	95,14	5,65	16,77	20228
	Average	**88,077**	**6,676**	**14,524**	**23853,3**
Urban / Road Transportation (Intermunicipal)	HR_9	108,18	1,67	64,81	6011
	HR_13	303,63	8,48	36,10	30264
	HR_18	348,75	8,11	43,03	29127
	HR_21	417,03	9,41	44,61	33631
	HR_26	196,01	8,61	22,91	30336
	HR_41	343,66	7,75	44,33	27863
	HR_46	364,03	8,57	42,58	30747
	HR_56	388,47	8,42	46,21	30225
	Average	**308,721**	**7,627**	**43,073**	**27275,5**

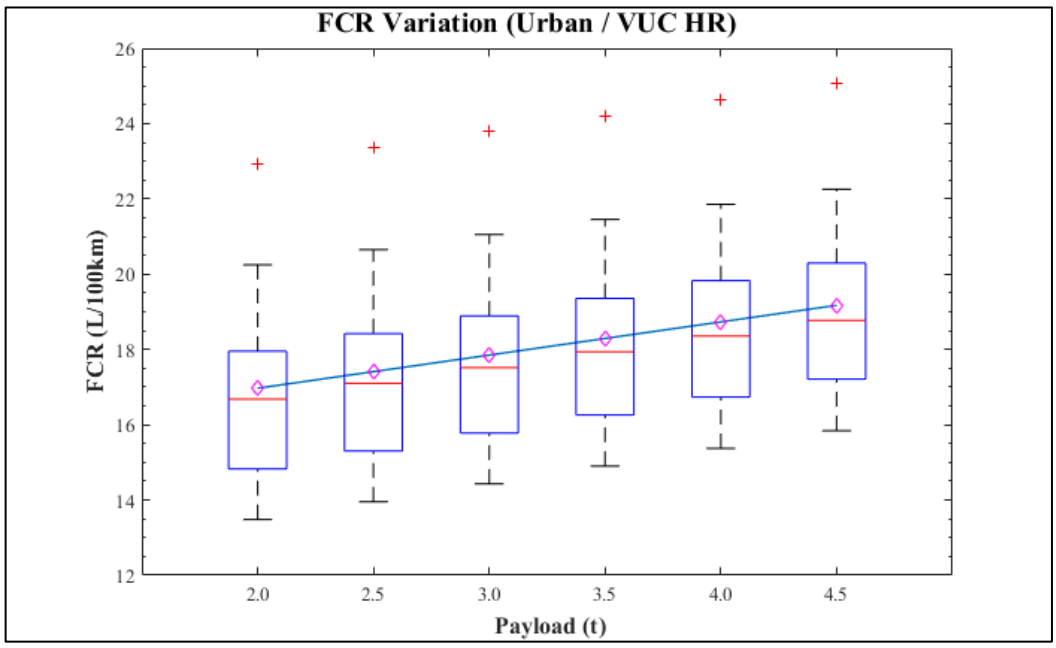

Figure 19 – Fuel consumption rate variation for a VUC running in an urban environment.

6.2 Algorithm Tuning

Gathered all the needed information to generate the instances based on São Paulo and the fuel consumption estimation, we could start the optimization and heuristics experiments. We also performed our tests using the CMT set of instances (Christofides et al., 1978) and the Golden set of instances (Golden et al., 1998), which are both used by Xiao et al. (2012) and Suzuki (2016), in a way that is possible to establish a direct benchmark for the PPRP.

In our experiments, we used version 8.5 of Gurobi optimization package running on a standard desktop personal computer (PC) and C++ environment to exactly solve the small problem instances and obtain valid lower bounds for both the PPRP and PPRP-TD. The PC has an Intel® Core™ i7-4690 CPU @ 3.5GHz with 16 GB of RAM. The same PC was also used to perform the algorithm tuning and run the GRASP FESA heuristic on all instances.

However, prior to running optimization and heuristics, we performed extensive experiments to tune and determine the best set of parameters σ and τ for our GRASP FESA heuristic. To this end, we carried out an experiment involving several different combinations of these two parameters that were run considering a sample of the SPZO, SP and CMT instances. The results we have obtained are presented in Table 11, in which values represent the percentage of improvement when compared to the worst case and bold values are the best values found.

Table 11 – Tuning objective results.

τ (# restarts)	σ (RCL size)							Average
	2	**3**	**4**	**5**	**6**	**7**	**8**	
10	+0,1889%	+0,1042%	+0,1307%	0,0000%	+0,1553%	+0,1943%	+0,0344%	+0,1154%
20	+0,7598%	+0,7353%	+0,7898%	+0,4741%	+0,4314%	+0,5415%	+0,5026%	+0,6049%
30	+0,8788%	+0,9487%	+0,9919%	+0,6448%	+0,5362%	+0,7501%	+0,6319%	+0,7689%
40	+0,9668%	+0,9987%	+1,0419%	+0,6960%	+0,5647%	+0,7786%	+0,6959%	+0,8204%
50	+0,9860%	+1,0201%	+1,0968%	+0,9367%	+0,7164%	+0,7827%	+0,7001%	+0,8913%
60	+0,9902%	+1,0374%	+1,1319%	+1,0028%	+0,7592%	+0,8242%	+0,7416%	+0,9268%
70	+1,0317%	+1,0727%	+1,1494%	+1,0126%	+0,8101%	+0,8659%	+0,8088%	+0,9645%
80	+1,0415%	+1,0838%	+1,1605%	+1,0707%	+0,8256%	+0,8850%	+0,8280%	+0,9850%
90	+1,0606%	+1,0979%	+1,1871%	+1,0865%	+0,9125%	+0,8985%	+0,8433%	+1,0123%
100	+1,0759%	+1,1072%	**+1,1964%**	+1,0957%	+0,9144%	+0,9097%	+0,8686%	**+1,0240%**
Average	+0,8980%	+0,9206%	**+0,9876%**	+0,8020%	+0,6626%	+0,7431%	+0,6655%	

These results show that our GRASP FESA heuristic is quite sensitive to the values of both σ (which sets the maximum size of the restricted candidate list) and τ (which sets the number of restarts). Though the highest number of restarts yielded the best results, the heuristic runtime is linear correlated to it (because the heuristic is able to explore a higher number of solutions), as evidenced by Table 12. The setting of σ, however, does not affect the total runtime.

Hereinafter, for our remaining experiments, we chose to use the parameter setting ($\sigma = 4$, $\tau = 70$), as it presented good results while still under a reasonable runtime. In real-world applications of the heurisic, the use of the highest possible number of restarts is encouraged if there is enough time available to do so.

Table 12 – Tuning runtime results.

τ	**Total Runtime (s)**
10	1339,6
20	2671,3
30	4181,0
40	5330,9
50	6635,1
60	7961,6
70	9310,2
80	10723,1
90	12259,1
100	13908,3

6.3 Results and discussion

As the MILP formulation for the PPRP-TD (Expressions 9–28) aims to minimize fuel consumption and uses an index k for used vehicles, an upper bound for the number of required vehicles in each instance is suitable. We follow Figliozzi (2012) and to obtain feasible upper bounds of good quality, we first used Gurobi in the PC to solve the problem by minimizing the number of used vehicles, thus using the objective function given by Expression (39) instead of using Expression (9). For each problem instance, we then used the best objective for this problem provided by Gurobi as an upper bound (i.e., the number of available vehicles) in the original formulation minimizing fuel consumption.

$$\text{minimize} \sum_{m \epsilon M} \sum_{k \in K} \sum_{j \in N} x_{0jk}^{m} \tag{39}$$

Detailed results are presented in the next subsections for both the PPRP and PPRP-TD.

6.3.1 PPRP results

Due to the fact that our GRASP-FESA solution method was designed as a two-stage algorithm (routing then scheduling), it is possible to evaluate its performance in the case of solving the PPRP only. Also, this evaluation allows us to compare the proposed method to existing algorithms from the literature, such as the String-Model-based Simulated Annealing with Hybrid exchange rule (SMSAH) presented by Xiao et al. (2012), and the Hybrid Simulated Annealing with Tabu Search (Hybrid SA-TS), a dual-objective approach proposed by Suzuki (2016).

To this end, we considered the same set of test instances used in these works, which includes 7 VRP instances taken from Christofides et al. (1978) (instances CMT-1 to CMT-5 and CMT-11, CMT-12), and 20 VRP instances taken from Golden et al. (1998) (instances Golden-1 to Golden-20). These two sets are simply referred to as CMT and Golden sets of instances.

In this way, for both the CMT and Golden sets, we used the same RCL size $\sigma = 4$ as for the SP and SPZO instances, but with only $\tau = 10$ restarts (instead of 70 adopted after the algorithm tuning), $\rho_0 = 1$ and $\rho^* = 2$, which are the same number of random seeds and FCR values used by Xiao et al. (2012) and Suzuki (2016). The results for the CMT and Golden sets are presented in Tables 13 and 14, respectively.

From Table 13, we can observe that GRASP-FESA is able to reach a final solution very quickly, but not always with the same quality of solutions provided by other available algorithms. However, for larger instances (e.g., the Golden set), it is possible to see that the runtime of our algorithm also increases substantially. In these cases, the use of a higher number of restarts (τ) might result in very long runtimes, what might not be suitable for real-life-sized instances faced by some practitioners.

Table 13 – PPRP results for the CMT set of instances.

Instance		SMSAH (Xiao et al, 2012)			Hybrid SA-TS (Suzuki, 2016)			GRASP-FESA		
Name	n	Min. Fuel (L)	Run-time (min)	Gap (%)	Min. Fuel (L)	Run-time (min)	Gap (%)	Min. Fuel (L)	Gap (%)	Run-time (min)
CMT-1	50	751,11	–	0,04%	751,11	1,09	0,00%	766,20	1,97%	0,00
CMT-2	75	1179,53	–	1,35%	1175,40	1,19	0,98%	1207,05	2,85%	0,00
CMT-3	100	1147,83	–	0,50%	1147,83	1,37	0,06%	1205,82	4,81%	0,01
CMT-4	150	1452,88	–	1,03%	1452,42	2,49	0,85%	1514,36	4,47%	0,09
CMT-5	199	1844,87	–	1,52%	1846,97	3,40	1,30%	1905,18	3,59%	0,33
CMT-11	120	1513,48	–	0,19%	1513,69	1,65	0,46%	1530,72	1,13%	0,04
CMT-12	100	1174,02	–	0,13%	1174,02	1,70	0,00%	1183,60	0,81%	0,01
		1294,82	**1,3**	**0,68%**	**1294,49**	**1,84**	**0,52%**	**1330,42**	**2,80%**	**0,07**

In addition to the CMT and Golden instances, we also tested the GRASP-FESA solution method against our new SP and SPZO sets of instances (in these cases using $\tau = 70$ as indicated in the previous tuning section). Because these are newly introduced instances, we provide the optimum results provided by Gurobi for the SPZO set of small instances, as well as the best available lower bounds (LB) and objectives found for the SP instances after a 2h time limit. Some of these lower bounds were also yielded by relaxing the MILP integrality constraints of the problem (Expression 22). The GRASP-FESA results for the SPZO and SP sets of instances are presented in Tables 15 and 16, respectively.

In the case of these São Paulo-based instances, our algorithm was able to achieve optimality for the majority of the SPZO instances, in much shorter times than Gurobi. In the case of the instances in which optimal solutions could not be obtained by GRASP-FESA, the algorithm was still able to provide good solutions that are close to the optimum solutions yielded by Gurobi, but in much shorter times (i.e., within fractions of seconds, in contrast to hundreds or tens of seconds in the case of optimization). In general, we could obtain very good solutions for these small SPZO instances.

Although Gurobi was able to achieve optimality for all SPZO instances, this was not the case for none of the larger SP instances, in which Gurobi failed to reach the optimum solutions. In fact, for instances with more than 100 customers, Gurobi could not find any feasible integer solutions within the imposed 2h time limit. In these cases, GRASP-FESA results are the only available, scoring an average gap of 20.42%. Although this value might seem high, it can be biased because of low quality lower bounds (e.g., calculated in simple ways such as relaxing the integrality constraints).

Table 14 – PPRP results for the Golden set of instances.

Instance			SMSAH (Xiao et al, 2012)			Hybrid SA-TS (Suzuki, 2016)			GRASP-FESA		
Name	**n**	**Capacity**	**Min. Fuel (L)**	**Runtime (min)**	**Gap (%)**	**Min. Fuel (L)**	**Runtime (min)**	**Gap (%)**	**Min. Fuel (L)**	**Gap (%)**	**Runtime (min)**
Golden-1	240	550	7683,52	–	0,70%	7666,4	5,63	0,39%	7877,71	2,76%	0,99
Golden-2	320	700	11172,71	–	0,39%	11166,57	7,14	0,37%	11717,30	4,83%	4,30
Golden-3	400	900	14497,64	–	0,58%	14494,3	11,19	1,09%	15097,90	4,08%	13,52
Golden-4	480	1000	18327,03	–	2,13%	18284,8	27,63	1,71%	18669,90	2,47%	35,55
Golden-5	200	900	8561,53	–	0,18%	8561,53	3,43	0,04%	8561,53	0,0%	0,45
Golden-6	280	900	11102,22	–	0,42%	11077,1	7,03	0,24%	11200,50	1,13%	2,18
Golden-7	360	900	13422,16	–	0,62%	13405,6	8,39	0,60%	13802,20	2,97%	7,85
Golden-8	440	900	15928,26	–	3,77%	15602,91	15,55	1,94%	16321,90	5,09%	22,86
Golden-9	255	1000	850,8	–	2,41%	845,24	4,18	1,27%	870,41	3,77%	1,76
Golden-10	323	1000	1083	–	2,37%	1074,94	6,89	1,78%	1109,98	4,06%	6,04
Golden-11	399	1000	1352,32	–	2,79%	1334,89	7,64	1,54%	1373,71	3,74%	16,33
Golden-12	483	1000	1630,81	–	3,60%	1629,83	13,29	2,79%	1660,14	3,54%	42,39
Golden-13	252	1000	1261,93	–	2,63%	1247,33	5,47	1,14%	1279,81	3,42%	0,98
Golden-14	320	1000	1595,48	–	2,45%	1584,24	6,64	1,68%	1628,48	3,87%	3,31
Golden-15	396	1000	1970,43	–	2,58%	1947,57	11,55	1,53%	2017,80	4,03%	9,85
Golden-16	480	1000	2391,12	–	2,81%	2380,04	14,08	2,04%	2430,46	3,68%	27,33
Golden-17	240	200	1027,21	–	1,53%	1029,26	3,93	1,48%	1057,08	3,70%	0,77
Golden-18	300	200	1462,31	–	1,64%	1453,21	5,26	1,01%	1502,37	3,76%	2,34
Golden-19	360	200	2007,62	–	2,05%	1987,13	6,79	1,30%	2064,78	4,45%	5,84
Golden-20	420	200	2687,85	–	2,40%	2668,24	19,44	1,78%	2740,21	3,86%	12,84
		Average	**6000,80**	**3,3**	**1,90%**	**5972,06**	**9,56**	**1,29%**	**6141,96**	**3,37%**	**10,88**

Table 15 – PPRP results for the small SPZO instances.

Instance			Gurobi				GRASP-FESA			
Vehicle Capacity	**# Customers**	**Instance Name**	**# Routes**	**Total Fuel (L)**	**Gap**	**Gurobi Runtime (s)**	**# Routes**	**Total Fuel (L)**	**Gap**	**Algorithm Runtime (s)**
250	5	SPZO_R_5_1	1	8,9203	**0,00%**	0,01	1	8,9203	**0,00%**	0,05
		SPZO_R_5_2	1	8,9400	**0,00%**	0,01	1	8,9400	**0,00%**	0,03
		SPZO_R_5_3	1	9,1819	**0,00%**	0,02	1	9,1819	**0,00%**	0,04
		SPZO_R_5_4	1	7,8506	**0,00%**	0,01	1	7,8506	**0,00%**	0,04
		SPZO_R_5_5	1	10,2192	**0,00%**	0,00	1	10,2192	**0,00%**	0,03
	10	SPZO_R_10_1	2	16,5573	**0,00%**	0,39	2	16,5573	**0,00%**	0,17
		SPZO_R_10_2	2	16,7758	**0,00%**	0,35	2	16,7758	**0,00%**	0,18
		SPZO_R_10_3	2	16,6372	**0,00%**	0,28	2	16,6372	**0,00%**	0,15
		SPZO_R_10_4	2	15,8438	**0,00%**	0,30	2	15,8438	**0,00%**	0,16
		SPZO_R_10_5	2	15,7927	**0,00%**	0,14	2	15,7927	**0,00%**	0,18
	15	SPZO_R_15_1	3	21,6428	**0,00%**	13,85	3	21,6428	**0,00%**	0,36
		SPZO_R_15_2	3	22,6844	**0,00%**	4,61	3	22,6844	**0,00%**	0,35
		SPZO_R_15_3	3	22,9178	**0,00%**	2,64	3	22,9907	0,32%	0,32
		SPZO_R_15_4	3	22,2569	**0,00%**	8,06	3	22,2569	**0,00%**	0,37
		SPZO_R_15_5	3	23,2017	**0,00%**	4,19	3	23,2766	0,32%	0,39
	20	SPZO_R_20_1	3	24,8112	**0,00%**	16,32	3	24,8112	**0,00%**	0,66
		SPZO_R_20_2	3	24,8696	**0,00%**	19,07	3	24,9941	0,50%	0,61
		SPZO_R_20_3	4	28,5908	**0,00%**	250,90	4	28,5908	**0,00%**	0,67
		SPZO_R_20_4	3	25,1219	**0,00%**	309,72	3	25,1219	**0,00%**	0,73
		SPZO_R_20_5	4	29,3792	**0,00%**	209,00	4	29,4899	0,38%	0,67
	25	SPZO_R_25_1	4	30,5360	**0,00%**	568,10	4	31,2924	2,42%	1,13
		SPZO_R_25_2	4	30,5080	**0,00%**	845,03	4	30,5509	0,14%	1,14
		SPZO_R_25_3	4	32,9306	**0,00%**	77,63	4	33,3260	1,19%	1,26
		SPZO_R_25_4	4	30,9734	**0,00%**	1843,59	4	30,9734	**0,00%**	1,02
		SPZO_R_25_5	4	30,9674	**0,00%**	501,41	4	31,2780	0,99%	1,12
		Total	**–**	**528,1106**	***0,00%***	**4675,61**	**–**	**529,9989**	***0,25%***	**11,85**

Table 16 – PPRP results for the large SP instances.

Instance			Gurobi					GRASP-FESA				
Vehicle Capacity	**# Customers**	**Instance Name**	**# Routes**	**Total Fuel (L)**	**Gurobi LB**	**Gap**	**Runtime (s)**	**# Routes**	**Total Fuel (L)**	**Best LB**	**Gap**	**Runtime (s)**
300	50	SP_R_50_1	6	88,5833	76,2634	**13,91%**	7200,17	6	91,0344	77,4241	14,95%	7,54
		SP_R_50_2	6	97,5664	77,9038	20,15%	7200,08	6	93,3081	80,0868	**14,17%**	6,88
		SP_R_50_3	6	92,2705	77,3037	16,22%	7200,07	6	90,3187	79,1304	**12,39%**	6,97
		SP_R_50_4	6	110,6570	85,7961	22,47%	7200,13	6	101,6890	87,5604	**13,89%**	6,67
		SP_R_50_5	6	108,4350	87,4096	19,39%	7200,07	6	102,8260	87,8156	**14,60%**	6,75
		SP_R_50_6	6	95,1599	78,5872	17,42%	7200,10	6	89,6319	81,1594	**9,45%**	6,92
		SP_R_50_7	6	102,9100	81,3327	20,97%	7200,11	6	98,3120	85,3987	**13,14%**	6,85
		SP_R_50_8	6	89,6274	72,7967	18,78%	7200,17	6	88,4252	74,7769	**15,43%**	7,09
		SP_R_50_9	6	97,8650	79,5713	18,69%	7200,12	6	94,8032	80,8388	**14,73%**	6,49
		SP_R_50_10	6	95,2981	79,4170	16,66%	7200,17	6	90,9804	80,9732	**11,00%**	6,57
	100	SP_R_100_1	–	–	132,5620	–	7200,03	12	165,4950	132,5620	**19,90%**	100,27
		SP_R_100_2	–	–	149,5150	–	7200,04	12	192,9160	149,5150	**22,50%**	105,17
		SP_R_100_3	–	–	134,4370	–	7200,04	12	167,0550	134,4370	**19,53%**	105,92
		SP_R_100_4	–	–	137,4090	–	7200,03	12	169,2840	137,4090	**18,83%**	103,99
		SP_R_100_5	–	–	142,8480	–	7200,03	13	176,3090	142,8480	**18,98%**	101,90
		SP_R_100_6	–	–	145,4190	–	7200,04	13	180,0940	145,4190	**19,25%**	102,87
		SP_R_100_7	–	–	145,0100	–	7200,18	12	177,9020	145,0100	**18,49%**	101,28
		SP_R_100_8	–	–	133,2468	–	7200,00	12	165,4590	133,2468	**19,47%**	110,69
	150	SP_R_150_1	–	–	–	–	7200,00	18	251,2410	197,1404	**21,53%**	735,84
		SP_R_150_2	–	–	–	–	7200,00	18	249,1510	196,0270	**21,32%**	723,59
		SP_R_150_3	–	–	–	–	7200,00	19	248,9690	197,4306	**20,70%**	720,13
		SP_R_150_4	–	–	–	–	7200,00	19	261,5400	205,8504	**21,29%**	686,83
		SP_R_150_5	–	–	–	–	7200,00	18	243,4480	195,9939	**19,49%**	708,70
		SP_R_150_6	–	–	–	–	7200,00	18	248,1780	193,9108	**21,87%**	718,72
	200	SP_R_200_1	–	–	–	–	7200,00	24	328,4470	256,4547	**21,92%**	2842,61
		SP_R_200_2	–	–	–	–	7200,00	24	317,4790	252,2750	**20,54%**	2845,05
		SP_R_200_3	–	–	–	–	7200,00	23	316,0180	251,9079	**20,29%**	2676,42
		SP_R_200_4	–	–	–	–	7200,00	23	313,2760	247,6200	**20,96%**	2661,39
		SP_R_200_5	–	–	–	–	7200,00	23	305,4910	241,9469	**20,80%**	2752,58
	250	SP_R_250_1	–	–	–	–	7200,00	30	388,3380	308,1191	**20,66%**	11386,40
		Total	**–**	**978,3726**	**1916,8283**	**18,47%**	**216001,58**	**–**	**941,3289**	**1935,6111**	**13,38%**	**30359,09**

6.3.2 PPRP-TD results

After solving the PPRP constructing all routes with the GRASP-FESA, the process of TD scheduling is started. In our experiments, we use a time-step $\varepsilon = 10$ minutes for the case of the SP and SPZO instances. In essence, this means that our 24h time horizon is discretized into $\left\lfloor {}^{\beta_0}/_{\varepsilon} \right\rfloor = 144$ equal parts. The time-table of scheduling is performed for each generated route, making it possible to obtain the best TD solution (i.e., with the least fuel consumption) within those 144 scheduled options.

Table 17 contains the PPRP-TD results for the small SPZO instances, allowing the comparison of the solutions found by Gurobi under a time limit of 2h and the solutions yielded by the GRASP-FESA with TD Scheduling. Gurobi was able to obtain the optimal solutions only for the instances with up to 20 customers, already taking long runtimes for the instances with 15 or more customers. Our solution method, however, is also able to achieve the optimal solutions for several of the smaller instances, although with almost instantaneous runtimes, solving every instance under 2 seconds. On average, both Gurobi and GRASP-FESA with TD scheduling are able to obtain similar gap results, but with a very large discrepancy in runtimes (i.e., of four orders of magnitude).

As for the large SP instances, Table 18 presents all the obtained results. Similar to the PPRP case, Gurobi couldn't obtain optimal solutions, not even being able to obtain feasible integer solutions for any of the instances. In fact, Gurobi could only obtain the lower bound values for the instances with 100 customers using the formulation given by Expressions (9–28). By relaxing the integrality constraints (22), Gurobi obtained some additional lower bounds, indicated in the *"Best LB"* column. In contrast to Gurobi, our metaheuristic solution method yielded solutions for every instance. Again, the gaps obtained might seem high, but they are probably due to the low-quality lower bounds obtained.

By analyzing the runtimes, it is possible to observe that the GRASP-FESA with TD scheduling also takes long times to run for the larger instances, mainly because of the constructive part of the solution method (i.e., FESA). This fact can be evidenced by the runtime comparison for the main components of the metaheuristic, presented in Figure 20 with a cumulative column chart, in which each column refers to an instance.

Table 17 – PPRP-TD results for the small SPZO instances.

Instance			Gurobi					GRASP-FESA with TD-Scheduling				
Vehicle Capacity	**# Customers**	**Instance Name**	**# Routes**	**Total Fuel (L)**	**Gurobi LB**	**Gap**	**CPLEX Runtime (s)**	**# Routes**	**Total Fuel (L)**	**Best LB**	**Gap**	**Runtime (s)**
250	5	SPZO_R_5_1	1	9,8799	9,8799	**0,00%**	0,18	1	9,8799	9,8799	**0,00%**	0,06
		SPZO_R_5_2	1	10,4995	10,4995	**0,00%**	0,11	1	10,4995	10,4995	**0,00%**	0,06
		SPZO_R_5_3	1	11,1008	11,1008	**0,00%**	0,23	1	11,1008	11,1008	**0,00%**	0,06
		SPZO_R_5_4	1	9,2425	9,2425	**0,00%**	0,12	1	9,2425	9,2425	**0,00%**	0,06
		SPZO_R_5_5	1	11,8280	11,8280	**0,00%**	0,08	1	11,8280	11,8280	**0,00%**	0,06
	10	SPZO_R_10_1	2	19,3644	19,3644	**0,00%**	8,26	2	19,3644	19,3644	**0,00%**	0,20
		SPZO_R_10_2	2	19,4907	19,4907	**0,00%**	14,20	2	19,4907	19,4907	**0,00%**	0,21
		SPZO_R_10_3	2	19,5821	19,5821	**0,00%**	6,19	2	19,5821	19,5821	**0,00%**	0,19
		SPZO_R_10_4	2	18,3862	18,3862	**0,00%**	12,70	2	18,3862	18,3862	**0,00%**	0,19
		SPZO_R_10_5	2	19,0469	19,0469	**0,00%**	4,43	2	19,0469	19,0469	**0,00%**	0,25
	15	SPZO_R_15_1	3	25,2266	25,2266	**0,00%**	476,14	3	25,2266	25,2266	**0,00%**	0,50
		SPZO_R_15_2	3	27,1954	27,1954	**0,00%**	1858,84	3	27,1954	27,1954	**0,00%**	0,41
		SPZO_R_15_3	3	27,5449	27,5437	**0,00%**	519,65	3	27,5449	27,5437	**0,00%**	0,43
		SPZO_R_15_4	3	26,4927	26,4927	**0,00%**	875,34	3	26,5807	26,4927	0,33%	0,41
		SPZO_R_15_5	3	27,3037	27,3037	**0,00%**	702,86	3	27,3037	27,3037	**0,00%**	0,43
	20	SPZO_R_20_1	3	30,0149	30,0149	**0,00%**	1500,91	3	30,0149	30,0149	**0,00%**	0,74
		SPZO_R_20_2	3	30,2789	30,2789	**0,00%**	3998,83	3	30,4872	30,2789	0,68%	0,68
		SPZO_R_20_3	4	33,9962	32,3486	**4,85%**	7200,14	4	33,9962	32,3486	**4,85%**	0,79
		SPZO_R_20_4	3	30,1733	28,5394	**5,42%**	7200,09	3	30,1733	28,5394	**5,42%**	0,81
		SPZO_R_20_5	4	35,3201	33,2947	**5,73%**	7200,14	4	35,4204	33,2947	6,00%	0,71
	25	SPZO_R_25_1	4	37,6974	33,8669	10,16%	7200,11	4	37,6550	33,8669	**10,06%**	1,24
		SPZO_R_25_2	4	36,8603	33,8635	8,13%	7200,45	4	36,7808	33,8635	**7,93%**	1,19
		SPZO_R_25_3	4	39,4603	37,1984	**5,73%**	7200,44	4	39,7898	37,1984	6,51%	1,34
		SPZO_R_25_4	4	37,8456	34,7522	**8,17%**	7200,24	4	37,9787	34,7522	8,50%	1,14
		SPZO_R_25_5	4	37,3768	34,3846	**8,01%**	7200,15	4	37,3768	34,3846	**8,01%**	1,27
		Total	**–**	**631,2081**	**610,7252**	***2,25%***	**67580,83**	**–**	**631,9454**	**610,7252**	***2,33%***	**13,44**

Table 18 – PPRP-TD results for the large SP instances.

Instance			Gurobi					GRASP-FESA with TD-Scheduling				
Vehicle Capacity	**# Customers**	**Instance Name**	**# Routes**	**Total Fuel (L)**	**Gurobi LB**	**Gap**	**Runtime (s)**	**# Routes**	**Total Fuel (L)**	**Best LB**	**Gap**	**Runtime (s)**
300	50	SP_R_50_1	–	–	87,6994	**–**	7200,12	6	111,4960	87,6994	**21,34%**	8,62
		SP_R_50_2	–	–	91,5240	–	7200,28	6	114,9370	91,5240	**20,37%**	8,31
		SP_R_50_3	–	–	90,1630	–	7200,26	6	110,3100	90,1630	**18,26%**	8,29
		SP_R_50_4	–	–	100,1860	–	7200,26	6	125,9100	100,1860	**20,43%**	7,20
		SP_R_50_5	–	–	103,1920	–	7205,66	6	127,9590	103,1920	**19,36%**	7,25
		SP_R_50_6	–	–	87,7079	–	7200,12	6	109,1800	87,7079	**19,67%**	7,36
		SP_R_50_7	–	–	95,3818	–	7200,29	6	119,7590	95,3818	**20,36%**	7,58
		SP_R_50_8	–	–	84,9708	–	7200,07	6	107,4460	84,9708	**20,92%**	7,73
		SP_R_50_9	–	–	92,6990	–	7200,24	6	117,5280	92,6990	**21,13%**	6,97
		SP_R_50_10	–	–	92,2227	–	7200,31	7	112,1850	92,2227	**17,79%**	7,29
	100	SP_R_100_1	–	–	–	–	7200,79	12	203,9060	156,6540	**23,17%**	113,59
		SP_R_100_2	–	–	–	–	7200,00	12	238,7800	178,0132	**25,45%**	102,91
		SP_R_100_3	–	–	–	–	7200,00	12	205,9460	158,5309	**23,02%**	103,53
		SP_R_100_4	–	–	–	–	7200,00	12	209,8970	162,9511	**22,37%**	104,12
		SP_R_100_5	–	–	–	–	7200,00	13	217,3040	170,0475	**21,75%**	102,05
		SP_R_100_6	–	–	–	–	7200,00	13	220,3050	172,9396	**21,50%**	103,00
		SP_R_100_7	–	–	–	–	7200,00	12	218,0010	170,9586	**21,58%**	102,73
		SP_R_100_8	–	–	–	–	7200,00	12	202,9290	159,6299	**21,34%**	105,46
	150	SP_R_150_1	–	–	–	–	7200,00	18	309,9910	237,3041	**23,45%**	692,19
		SP_R_150_2	–	–	–	–	7200,00	17	306,8590	236,5345	**22,92%**	695,89
		SP_R_150_3	–	–	–	–	7200,00	19	307,3630	–	–	682,47
		SP_R_150_4	–	–	–	–	7200,00	19	322,5630	–	–	708,09
		SP_R_150_5	–	–	–	–	7200,00	18	298,8350	237,0791	**20,67%**	713,11
		SP_R_150_6	–	–	–	–	7200,00	18	304,1900	234,1046	**23,04%**	663,10
	200	SP_R_200_1	–	–	–	–	7200,00	24	403,6290	–	–	2667,37
		SP_R_200_2	–	–	–	–	7200,00	24	389,4320	–	–	2619,13
		SP_R_200_3	–	–	–	–	7200,00	24	389,3810	–	–	3005,86
		SP_R_200_4	–	–	–	–	7200,00	23	387,7250	–	–	2600,81
		SP_R_200_5	–	–	–	–	7200,00	23	376,4720	–	–	2636,91
	250	SP_R_250_1	–	–	–	–	7200,00	30	479,0450	–	–	7777,86
		Total	**–**	**–**	**925,7466**	**–**	**216008,40**	**–**	**–**	**925,7466**	**–**	**26376,76**

The GRASP-FESA runtime is composed of the red and blue columns (Randomized FESA plus the local search procedures), while the TD scheduling part are the columns in green. By observing this chart, it is possible to note that when there is an increase in the number of customers of the instances, the composition of the total solution method runtime changes from a majority of runtime being consumed by the TD scheduling to a majority (almost 100%) of the runtime being computed in the R-FESA stage.

This runtime analysis suggests the requirement of the R-FESA to be more efficient when solving large instances. Adjustments or simplifications to the greedy heuristic may allow a larger portion of the solution space be explored by the local search procedures or by the TD scheduling with lower ε values.

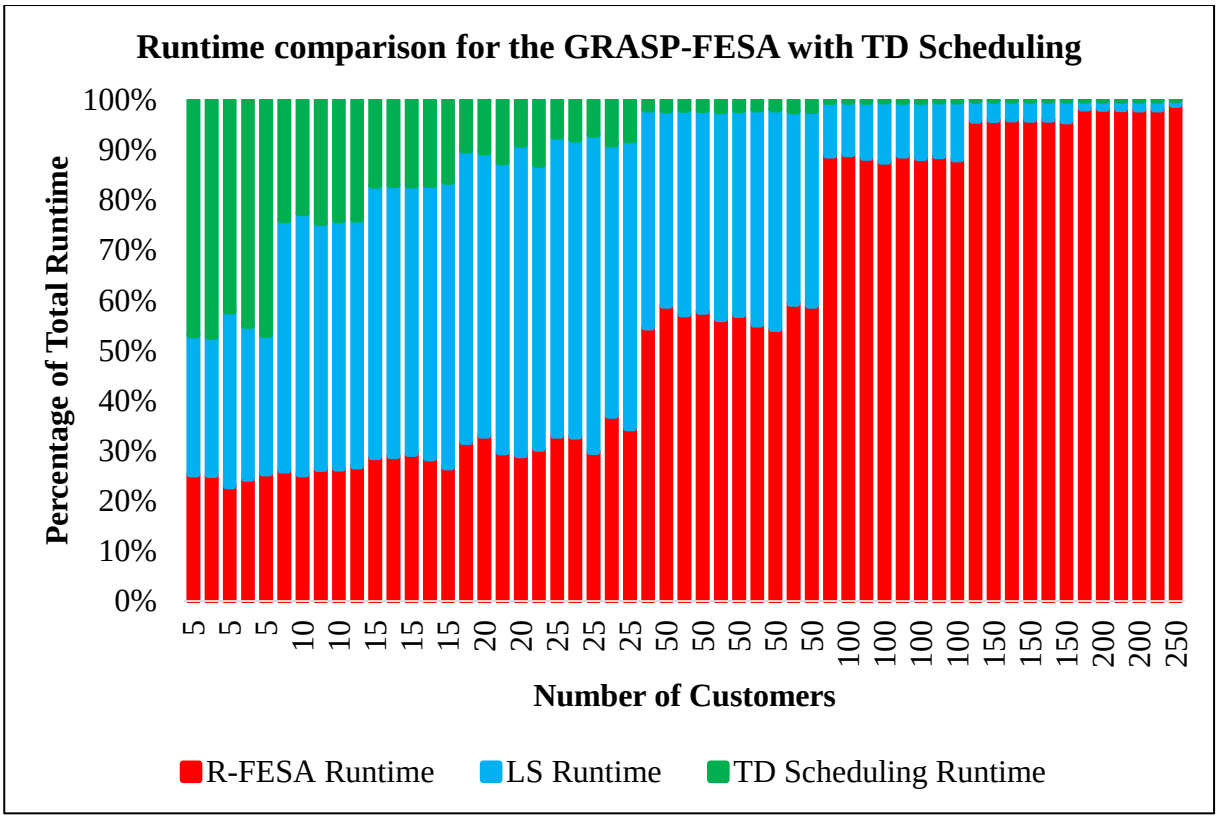

Figure 20 – Runtime comparison for all SP and SPZO instances.

7 Conclusion

In this research, we have addressed the Practical Pollution-Routing Problem with Time-Dependent (PPRP-TD) speeds, a new practical variant of the Pollution-Routing Problem that considers mainly the effects of the payload and time-dependent speeds to minimize fuel consumption of the vehicles. The practical aspect of our PPRP-TD regards the number of parameters and user inputs that are needed when compared to other approaches. Fuel consumption is estimated using an FCR linear function, which can be modeled to a specific fleet using real GPS data. In our case, we use GPS data from light-duty vehicles delivering in São Paulo and model the FCR function is modeled using and the Comprehensive Modal Emission Modal (CMEM).

We propose a Greedy Randomized Adaptive Search Procedure FCR-based Extended Savings Algorithm (GRASP-FESA) to solve both the PPRP and PPRP-TD. In the case of the PPRP-TD, extensive time-dependent scheduling is performed on top of the GRASP-FESA solution to obtain a valid final solution that considers time-dependent travel speeds and the effect of fluctuating speeds that occurs during congestion periods.

The solution method is a metaheuristic that relies on a randomized greed heuristic based on the classical savings algorithm. We improved the savings algorithm by considering the fuel consumption in the savings calculation and also by evaluating different types of merging routes in the process. In our case, routes can also be merged by insertion moves and not only by concatenations such as in the original algorithm.

Our solution method is then tested against two sets of instances from the literature in the case of the PPRP, and also against two newly introduced sets of instances based on real retail distribution in São Paulo for both the PPRP and PPRP-TD. Regarding the MILP and heuristic results, we could obtain good solutions using GRASP-FESA (with time-dependent scheduling), although within some high runtimes for the larger instances.

The two new sets of instances were also modeled using time-dependent travel times for all pairs of customers and fuel consumption rate functions based on real GPS tracks from trucks delivering in São Paulo. We also obtained general time-dependent speed ratios for the 24 hours of the day in the SP network. In this aspect, we shed some light in the effects of directions in

urban logistics problems. Although being important, we do not consider this topic in our proposed problem and solution method, thus leaving it for a future research agenda.

Another point that needs further attention is the consideration of other neighborhood operators to our solution method, such as perturbation moves (e.g., shaking operators from the VNS literature (Hansen and Mladenović, 2001); and ruin and recreate operators from the ALNS literature (Franceschetti et al., 2017)) or even different local search procedures. A simplification to our greedy heuristic is also desired, in order to generate solutions more quickly, thus allowing the local searches and final method to further explore the solution space. In the case of the PPRP-TD, a more robust method for the TD scheduling could provide better solutions, such as dynamic programming approaches (e.g., Xiao and Konak, 2017) or even the development of a conjunct routing and scheduling solution method, in which the scheduling part and the construction of routes are considered at the same time.

In additions to these changes to the solution method, the use of a technique to generate better lower bounds should allow for a better comparison of solution methods. Better lower bounds makes possible to use more accurate gap values for the results evaluation.

References

Alinaghian, M., Naderipour, M., 2016. A novel comprehensive macroscopic model for time-dependent vehicle routing problem with multi-alternative graph to reduce fuel consumption: A case study. Comput. Ind. Eng. 99, 210–222. https://doi.org/10.1016/j.cie.2016.07.029

Anbuudayasankar, S.P., Ganesh, K., Lenny Koh, S.C., Ducq, Y., 2012. Modified savings heuristics and genetic algorithm for bi-objective vehicle routing problem with forced backhauls. Expert Syst. Appl. 39, 2296–2305. https://doi.org/10.1016/j.eswa.2011.08.009

Androutsopoulos, K.N., Zografos, K.G., 2017. An integrated modelling approach for the bicriterion vehicle routing and scheduling problem with environmental considerations. Transp. Res. Part C Emerg. Technol. 82, 180–209. https://doi.org/10.1016/j.trc.2017.06.013

Barth, M., Younglove, T., Scora, G., 2005. Development of a Heavy-Duty Diesel Modal Emissions and Fuel Consumption Model, California PATH Research Report. Berkeley, CA. https://doi.org/10.1016/j.trd.2009.01.004

Bektaş, T., Demir, E., Laporte, G., 2016. Green Vehicle Routing, in: Psaraftis, H.N. (Ed.), Green Transportation Logistics: The Quest for Win-Win Solutions. Springer International Publishing, pp. 243–265. https://doi.org/10.1007/978-3-319-17175-3_7

Bektas, T., Laporte, G., 2011. The Pollution-Routing Problem. Transp. Res. Part B Methodol. 45, 1232–1250. https://doi.org/10.1016/j.trb.2011.02.004

Bigazzi, A., Bertini, R., 2009. Adding Green Performance Metrics to a Transportation Data Archive. Transp. Res. Rec. J. Transp. Res. Board 2121, 30–40. https://doi.org/10.3141/2121-04

Bowyer, D.P., Akçelik, R., Biggs, D.C., 1985. Guide to Fuel Consumption Analysis for Urban Traffic Management. Vermont South, Australia.

Braekers, K., Ramaekers, K., Van Nieuwenhuyse, I., 2016. The vehicle routing problem: State of the art classification and review. Comput. Ind. Eng. 99, 300–313. https://doi.org/10.1016/j.cie.2015.12.007

Çatay, B., 2010. A new saving-based ant algorithm for the Vehicle Routing Problem with Simultaneous Pickup and Delivery. Expert Syst. Appl. 37, 6809–6817. https://doi.org/10.1016/j.eswa.2010.03.045

Christofides, N., Mingozzi, A., Toth, P., 1978. The Vehicle Routing Problem, in: Christofides, N., Mingozzi, A., Toth, P., Sandi, C. (Eds.), Combinatorial Optimization. Wiley, Chichester, pp. 315–338.

Çimen, M., Soysal, M., 2017. Time-dependent green vehicle routing problem with stochastic vehicle speeds: An approximate dynamic programming algorithm. Transp. Res. Part D Transp. Environ. 54, 82–98. https://doi.org/10.1016/j.trd.2017.04.016

CISLOG, 2015. Avaliação do projeto-piloto de entregas noturnas no município de São Paulo. Série Cadernos Técnicos vol. 18. ANTP, São Paulo.

Clarke, G., Wright, J.W., 1964. Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. Oper. Res. 12, 568–581. https://doi.org/10.1287/opre.12.4.568

Cunha, C.B., Yoshizaki, H.T.Y., Almeida, F.G.V., Hino, C.M., Kako, I.S., Laranjeiro, P., Arbex, R.O., 2016. Truck GPS data for in-depth characterization of urban logistics in the Megacity of São Paulo, Brazil. Proc. 2016 World Conf. Transp. Res.

Cunha, C.B., Yoshizaki, H.T.Y., Bartholomeu, D.B., 2017. Emissão de gases de efeito estufa (GEE) no transporte de cargas, 1st ed. Atlas, São Paulo.

DEFRA – Department for Environment Food & Rural Affairs, 2010. Guidelines to Defra / DECC's GHG Conversion Factors for Company Reporting. London, United Kingdom.

Dekker, R., Bloemhof, J., Mallidis, I., 2012. Operations Research for green logistics – An overview of aspects, issues, contributions and challenges. Eur. J. Oper. Res. 219, 671–679. https://doi.org/10.1016/j.ejor.2011.11.010

Demir, E., Bektas, T., Laporte, G., 2014a. A review of recent research on green road freight transportation. Eur. J. Oper. Res. 237, 775–793. https://doi.org/10.1016/j.ejor.2013.12.033

Demir, E., Bektas, T., Laporte, G., 2011. A comparative analysis of several vehicle emission models for road freight transportation. Transp. Res. Part D Transp. Environ. 16, 347–357. https://doi.org/10.1016/j.trd.2011.01.011

Demir, E., Bektaş, T., Laporte, G., 2014b. The bi-objective Pollution-Routing Problem. Eur.

J. Oper. Res. 232, 464–478. https://doi.org/10.1016/j.ejor.2013.08.002

Demir, E., Bektaş, T., Laporte, G., 2012. An adaptive large neighborhood search heuristic for the Pollution-Routing Problem. Eur. J. Oper. Res. 223, 346–359. https://doi.org/10.1016/j.ejor.2012.06.044

Eglese, R., Maden, W., Slater, A., 2006. A Road Timetable™ to aid vehicle routing and scheduling. Comput. Oper. Res. 33, 3508–3519. https://doi.org/10.1016/j.cor.2005.03.029

Ehmke, J.F., Campbell, A.M., Thomas, B.W., 2016. Vehicle routing to minimize time-dependent emissions in urban areas. Eur. J. Oper. Res. 251, 478–494. https://doi.org/10.1016/j.ejor.2015.11.034

Eksioglu, B., Vural, A.V., Reisman, A., 2009. The vehicle routing problem: A taxonomic review. Comput. Ind. Eng. 57, 1472–1483. https://doi.org/10.1016/j.cie.2009.05.009

Erdoğan, S., Miller-Hooks, E., 2012. A Green Vehicle Routing Problem. Transp. Res. Part E Logist. Transp. Rev. 48, 100–114. https://doi.org/10.1016/j.tre.2011.08.001

Feo, T.A., Resende, M.G.C., 1995. Greedy Randomized Adaptive Search Procedures. J. Glob. Optim. 6, 109–133. https://doi.org/10.1007/BF01096763

Feo, T.A., Resende, M.G.C., 1989. A probabilistic heuristic for a computationally difficult set covering problem. Oper. Res. Lett. 8, 67–71. https://doi.org/10.1016/0167-6377(89)90002-3

Figliozzi, M., 2010. Vehicle Routing Problem for Emissions Minimization. Transp. Res. Rec. J. Transp. Res. Board 2197, 1–7. https://doi.org/10.3141/2197-01

Figliozzi, M.A., 2012. The time dependent vehicle routing problem with time windows: Benchmark problems, an efficient solution algorithm, and solution characteristics. Transp. Res. Part E Logist. Transp. Rev. 48, 616–636. https://doi.org/10.1016/j.tre.2011.11.006

Figliozzi, M.A., 2011. The impacts of congestion on time-definitive urban freight distribution networks CO2 emission levels: Results from a case study in Portland, Oregon. Transp. Res. Part C Emerg. Technol. 19, 766–778. https://doi.org/10.1016/j.trc.2010.11.002

Figliozzi, M.A., 2010. The impacts of congestion on commercial vehicle tour characteristics and costs. Transp. Res. Part E Logist. Transp. Rev. 46, 496–506.

https://doi.org/10.1016/j.tre.2009.04.005

Franceschetti, A., Demir, E., Honhon, D., Van Woensel, T., Laporte, G., Stobbe, M., 2017. A metaheuristic for the time-dependent pollution-routing problem. Eur. J. Oper. Res. 259, 972–991. https://doi.org/10.1016/j.ejor.2016.11.026

Franceschetti, A., Honhon, D., Van Woensel, T., Bektaş, T., Laporte, G., 2013. The time-dependent pollution-routing problem. Transp. Res. Part B Methodol. 56, 265–293. https://doi.org/10.1016/j.trb.2013.08.008

Gendreau, M., Ghiani, G., Guerriero, E., 2015. Time-dependent routing problems: A review. Comput. Oper. Res. 64, 189–197. https://doi.org/10.1016/j.cor.2015.06.001

Golden, B.L., Wasil, E.A., Kelly, J.P., Chao, I.-M., 1998. The Impact of Metaheuristics on Solving the Vehicle Routing Problem: Algorithms, Problem Sets, and Computational Results, in: Crainic, T.G., Laporte (Eds.), Fleet Management and Logistics. Springer US, Boston, MA, pp. 33–56. https://doi.org/10.1007/978-1-4615-5755-5_2

Google Website. <https://developers.google.com/maps/documentation/> [Accessed 30 September 2017].

Guo, J., Liu, C., 2017. Time-Dependent Vehicle Routing of Free Pickup and Delivery Service in Flight Ticket Sales Companies Based on Carbon Emissions. J. Adv. Transp. 2017, 1–14. https://doi.org/10.1155/2017/1918903

Hansen, P., Mladenović, N., 2001. Variable neighborhood search: Principles and applications. Eur. J. Oper. Res. 130, 449–467. https://doi.org/10.1016/S0377-2217(00)00100-4

Heidelberg, I., Berne, Hannover, 2014. Ecological Transport Information Tool for Worldwide Transports: Methodology and Data Update.

Hickman, J., Hassel, D., Jourmard, R., Samaras, Z., Sorenson, S., 1999. Methodology for calculating transport emissions and energy consumption.

Holguín-Veras, J., Encarnación, T., González-Calderón, C.A., Winebrake, J., Wang, C., Kyle, S., Herazo-Padilla, N., Kalahasthi, L., Adarme, W., Cantillo, V., Yoshizaki, H., Garrido, R., 2016. Direct impacts of off-hour deliveries on urban freight emissions. Transp. Res. Part D Transp. Environ. https://doi.org/10.1016/j.trd.2016.10.013

Huang, Y., Zhao, L., Van Woensel, T., Gross, J.-P., 2017. Time-dependent vehicle routing problem with path flexibility. Transp. Res. Part B Methodol. 95, 169–195.

https://doi.org/10.1016/j.trb.2016.10.013

Ichoua, S., Gendreau, M., Potvin, J.-Y., 2003. Vehicle dispatching with time-dependent travel times. Eur. J. Oper. Res. 144, 379–396. https://doi.org/10.1016/S0377-2217(02)00147-9

Jabali, O., Van Woensel, T., de Kok, A.G., 2012. Analysis of Travel Times and CO 2 Emissions in Time-Dependent Vehicle Routing. Prod. Oper. Manag. 21, 1060–1074. https://doi.org/10.1111/j.1937-5956.2012.01338.x

Jaehn, F., 2016. Sustainable Operations. Eur. J. Oper. Res. 253, 243–264. https://doi.org/10.1016/j.ejor.2016.02.046

Juan, A., Mendez, C., Faulin, J., de Armas, J., Grasman, S., 2016. Electric Vehicles in Logistics and Transportation: A Survey on Emerging Environmental, Strategic, and Operational Challenges. Energies 9, 86. https://doi.org/10.3390/en9020086

Kouridis, C., Gkatzoflias, D., Kioutsioukis, I., Ntziachristos, L., Pastorello, C., Dilara, P., 2010. Uncertainty estimates and guidance for road transport emission calculations. Ispra, Italy. https://doi.org/10.2788/78236

Kuo, Y., 2010. Using simulated annealing to minimize fuel consumption for the time-dependent vehicle routing problem. Comput. Ind. Eng. 59, 157–165. https://doi.org/10.1016/j.cie.2010.03.012

Labadie, N., Prins, C., Prodhon, C., 2016. Metaheuristics for Vehicle Routing Problems. John Wiley & Sons, Inc., Hoboken, NJ, USA. https://doi.org/10.1002/9781119136767

Lin, C., Choy, K.L., Ho, G.T.S., Chung, S.H., Lam, H.Y., 2014. Survey of Green Vehicle Routing Problem: Past and future trends. Expert Syst. Appl. 41, 1118–1138. https://doi.org/10.1016/j.eswa.2013.07.107

Liu, W.-Y., Lin, C.-C., Chiu, C.-R., Tsao, Y.-S., Wang, Q., 2014. Minimizing the Carbon Footprint for the Time-Dependent Heterogeneous-Fleet Vehicle Routing Problem with Alternative Paths. Sustainability 6, 4658–4684. https://doi.org/10.3390/su6074658

Maden, W., Eglese, R., Black, D., 2010. Vehicle routing and scheduling with time-varying data: A case study. J. Oper. Res. Soc. 61, 515–522. https://doi.org/10.1057/jors.2009.116

Malandraki, C., Daskin, M.S., 1992. Time Dependent Vehicle Routing Problems: Formulations, Properties and Heuristic Algorithms. Transp. Sci. 26, 185–200. https://doi.org/10.1287/trsc.26.3.185

Naderipour, M., Alinaghian, M., 2016. Measurement, evaluation and minimization of CO 2 , NO x , and CO emissions in the open time dependent vehicle routing problem. Measurement 90, 443–452. https://doi.org/10.1016/j.measurement.2016.04.043

National Atmospheric Emissions Inventory, 2015. < http://naei.beis.gov.uk/data/ef-transport > [Accessed 12 September 2017].

Norouzi, N., Sadegh-Amalnick, M., Tavakkoli-Moghaddam, R., 2017. Modified particle swarm optimization in a time-dependent vehicle routing problem: minimizing fuel consumption. Optim. Lett. 11, 121–134. https://doi.org/10.1007/s11590-015-0996-y

Palmer, A., 2007. The development of an integrated routing and carbon dioxide emissions model for goods vehicles. Cranfield University.

Paschoal, A.O. de O., Almeida, F.G.V. de, Yoshizaki, H.T.Y., 2013. Análise comparativa de modelos de estimativa de consumo de combustível para veículos de carga, in: XXVII ANPET - Congresso Nacional de Pesquisa e Ensino Em Transporte. Belém, PA.

Paschoal, A.O. de O., Zancopé, A.L.W., Andrade, F.S., Almeida, F.G.V., Silva, R.J.M., 2017. Emissão de gases de efeito estufa - GEE, in: Cunha, C.B., Yoshizaki, H.T.Y., Bartholomeu, D.B. (Eds.), Emissão de Gases de Efeito Estufa (GEE) No Transporte de Cargas: Modelos e Aplicações No Brasil. Atlas, São Paulo, pp. 49–124.

Pelletier, S., Jabali, O., Laporte, G., 2016. Goods Distribution with Electric Vehicles: Review and Research Perspectives. Transp. Sci. 50, 3–22. https://doi.org/10.1287/trsc.2015.0646

Qian, J., Eglese, R., 2016. Fuel emissions optimization in vehicle routing problems with time-varying speeds. Eur. J. Oper. Res. 248, 840–848. https://doi.org/10.1016/j.ejor.2015.09.009

Qian, J., Eglese, R., 2014. Finding least fuel emission paths in a network with time-varying speeds. Networks 63, 96–106. https://doi.org/10.1002/net.21524

RACELOGIC Website. <https://www.vboxmotorsport.co.uk/index.php/en/products/video-loggers/video-vbox-pro> [Accessed 25 May 2018].

Saberi, M., Verbas, İ.Ö., 2012. Continuous Approximation Model for the Vehicle Routing Problem for Emissions Minimization at the Strategic Level. J. Transp. Eng. 138, 1368–1376. https://doi.org/10.1061/(ASCE)TE.1943-5436.0000442

Segerstedt, A., 2014. A simple heuristic for vehicle routing – A variant of Clarke and

Wright's saving method. Int. J. Prod. Econ. 157, 74–79. https://doi.org/10.1016/j.ijpe.2013.09.017

Solomon, M.M., 1987. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. Oper. Res. 35, 254–265. https://doi.org/10.1287/opre.35.2.254

Soysal, M., Bloemhof-Ruwaard, J.M., Bektaş, T., 2015. The time-dependent two-echelon capacitated vehicle routing problem with environmental considerations. Int. J. Prod. Econ. 164, 366–378. https://doi.org/10.1016/j.ijpe.2014.11.016

Soysal, M., Çimen, M., 2017. A Simulation Based Restricted Dynamic Programming approach for the Green Time Dependent Vehicle Routing Problem. Comput. Oper. Res. 88, 297–305. https://doi.org/10.1016/j.cor.2017.06.023

Srivatsa Srinivas, S., Gajanand, M.S., 2017. Vehicle routing problem and driver behaviour: a review and framework for analysis. Transp. Rev. 37, 590–611. https://doi.org/10.1080/01441647.2016.1273276

Stanojević, M., Stanojević, B., Vujošević, M., 2013. Enhanced savings calculation and its applications for solving capacitated vehicle routing problem. Appl. Math. Comput. 219, 10302–10312. https://doi.org/10.1016/j.amc.2013.04.002

Suzuki, Y., 2016. A dual-objective metaheuristic approach to solve practical pollution routing problem. Int. J. Prod. Econ. 176, 143–153. https://doi.org/10.1016/j.ijpe.2016.03.008

Toth, P., Vigo, D., 2014. Vehicle Routing, 2nd ed, MOS-SIAM Series on Optimization. Society for Industrial and Applied Mathematics, Philadelphia, PA. https://doi.org/10.1137/1.9781611973594

Turkensteen, M., 2017. The accuracy of carbon emission and fuel consumption computations in green vehicle routing. Eur. J. Oper. Res. 262, 647–659. https://doi.org/10.1016/j.ejor.2017.04.005

Wen, L., Eglese, R., 2015. Minimum cost VRP with time-dependent speed data and congestion charge. Comput. Oper. Res. 56, 41–50. https://doi.org/10.1016/j.cor.2014.10.007

Xiao, Y., Konak, A., 2017. A genetic algorithm with exact dynamic programming for the green vehicle routing & scheduling problem. J. Clean. Prod. 167, 1450–1463. https://doi.org/10.1016/j.jclepro.2016.11.115

Xiao, Y., Konak, A., 2016. The heterogeneous green vehicle routing and scheduling problem with time-varying traffic congestion. Transp. Res. Part E Logist. Transp. Rev. 88, 146–166. https://doi.org/10.1016/j.tre.2016.01.011

Xiao, Y., Konak, A., 2015. A simulating annealing algorithm to solve the green vehicle routing & scheduling problem with hierarchical objectives and weighted tardiness. Appl. Soft Comput. 34, 372–388. https://doi.org/10.1016/j.asoc.2015.04.054

Xiao, Y., Zhao, Q., Kaku, I., Xu, Y., 2012. Development of a fuel consumption optimization model for the capacitated vehicle routing problem. Comput. Oper. Res. 39, 1419–1431. https://doi.org/10.1016/j.cor.2011.08.013

Yao, E., Lang, Z., Yang, Y., Zhang, Y., 2015. Vehicle routing problem solution considering minimising fuel consumption. IET Intell. Transp. Syst. 9, 523–529. https://doi.org/10.1049/iet-its.2015.0027