European Journal of Operational Research 276 (2019) 637–655

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Innovative Applications of O.R.

A co-evolutionary matheuristic for the car rental capacity-pricing stochastic problem

Beatriz B. Oliveira[a,*], Maria Antónia Carravilla[a], José F. Oliveira[a], Alysson M. Costa[b]

[a] INESC TEC, Faculty of Engineering, University of Porto, Portugal
[b] School of Mathematics and Statistics, The University of Melbourne, Australia

ARTICLE INFO

Article history:
Received 2 March 2018
Accepted 7 January 2019
Available online 11 January 2019

Keywords:
Revenue management
Pricing
Car rental fleet management
Genetic algorithms
Stochastic programming

ABSTRACT

When planning a selling season, a car rental company must decide on the number and type of vehicles in the fleet to meet demand. The demand for the rental products is uncertain and highly price-sensitive, and thus capacity and pricing decisions are interconnected. Moreover, since the products are rentals, capacity "returns". This creates a link between capacity with fleet deployment and other tools that allow the company to meet demand, such as upgrades, transferring vehicles between locations or temporarily leasing additional vehicles.

We propose a methodology that aims to support decision-makers with different risk profiles plan a season, providing good solutions and outlining their ability to deal with uncertainty when little information about it is available. This matheuristic is based on a co-evolutionary genetic algorithm, where parallel populations of solutions and scenarios co-evolve. The fitness of a solution depends on the risk profile of the decision-maker and its performance against the scenarios, which is obtained by solving a mathematical programming model. The fitness of a scenario is based on its contribution in making the scenario population representative and diverse. This is measured by the impact the scenarios have on the solutions.

Computational experiments show the potential of this methodology regarding the quality of the solutions obtained and the diversity and representativeness of the set of scenarios generated. Its main advantages are that no information regarding probability distributions is required, it supports different decision-making risk profiles, and it provides a set of good solutions for an innovative complex application.

© 2019 Elsevier B.V. All rights reserved.

1. Introduction

When planning a selling season, a car rental company must decide on the fleet size and mix, i.e., the capacity it will have to meet demand throughout the season and rental locations. The demand is uncertain and highly price-sensitive. Therefore, the prices charged by a company are connected with and should influence the capacity decisions. Capacity decisions are also connected with other instruments that allow the company to "meet" its demand, which range from offering upgrades to transferring vehicles between locations or temporarily leasing additional vehicles.

The goal of this work is to provide decision-makers with profitable solutions to capacity and pricing decisions, assessing and increasing their ability to deal with the different realizations of uncertainty, represented by scenarios, when little information regarding those is available. The methodology developed is based on a co-evolutionary genetic algorithm, where parallel populations of solutions and scenarios co-evolve, depending on each other for the fitness evaluation of their individuals. On the one hand, this method aims at obtaining a representative and diverse population of scenarios, measured according to the impact they have on the population of solutions. On the other hand, the solutions evolve according to different decision-making risk profiles that assess its performance against the population of scenarios.

1.1. Previous works

This work deals with the integration of capacity and pricing decisions under uncertainty within the context of the car rental business. In this section, the relevance of the application and methodological scope of the work will be discussed. Firstly, the recently growing body of research on car rental fleet management and pricing will be briefly reviewed. This is an innovative and different application because the capacity is *rented* rather than sold.

* Corresponding author.
E-mail addresses: beatriz.b.oliveira@inescporto.pt, beatriz.oliveira@fe.up.pt (B.B. Oliveira).

https://doi.org/10.1016/j.ejor.2019.01.015
0377-2217/© 2019 Elsevier B.V. All rights reserved.

However, previous works that tackled the integration of pricing and capacity, although not directly applicable, can bring relevant insights to this problem. A stochastic approach to the problem is considered, where the uncertainty is represented by scenarios. Stochastic problems with similar characteristics are briefly reviewed regarding methodological approaches. Moreover, previous fundamental works that laid the foundation for the methodological idea developed in this paper will be presented.

1.1.1. Car rental fleet management and pricing

The car rental fleet management problem is initially structured in Pachon, Iakovou, Ip, and Aboudi (2003) and Pachon, Iakovou, and Ip (2006). Fink and Reiners (2006) extends the operational issues within fleet management and deployment, considering essential and realistic practical needs. In Oliveira, Carravilla, and Oliveira (2017c), the link with revenue management issues is introduced, and the body of research developed in this field is reviewed and structured. Existing gaps and relevant future research directions are discussed, including the integration of pricing and/or capacity allocation (revenue management issues) with operational decisions related to fleet size/mix and deployment. The need to consider uncertainty in demand in order to approximate the model to reality is also highlighted.

In a previous paper – Oliveira, Carravilla, and Oliveira (2018) – we tackled the first research direction. A mathematical model for the deterministic integration of dynamic pricing and capacity decisions was proposed. Due to the complexity of the problem, a matheuristic was proposed. This matheuristic is based on a decomposition of the problem, where the price decisions are directly encoded in the chromosomes, and the remaining decisions and the fitness of the full solution are obtained by solving a mathematical programming model. Moreover, some performance-boosting initial population generation procedures were proposed.

In this work, we propose to tackle the even more complex problem that arises when uncertainty is incorporated. Moreover, additional realistic requirements (such as price hierarchy) are included, and demand is modeled considering its relationship with competitor prices.

1.1.2. Integration of capacity, inventory and pricing decisions

Pricing decisions have often been tackled independently of capacity and inventory decisions. A recent and growing body of research on the integration of these topics has been arising.

Den Boer (2015) presents an interesting and thorough literature review on the topic of dynamic pricing, primarily focused on learning processes. Following the structure proposed by the author, the car rental pricing problem herein considered can be seen as a dynamic pricing problem with inventory effects, more specifically "jointly determining selling prices and inventory–procurement". In Gallego and van Ryzin (1994), the dynamic pricing problem for inventories with price-sensitive and stochastic demand is tackled, including an extension where the initial stock is considered as a decision variable. The *rental* facet of the problem at hand hinders the direct application of the insights drawn. Focusing on perishable assets, a dynamic pricing problem under competition is studied in Gallego and Hu (2014). Here, the dynamics of an oligopoly are considered, dealing with substitutability among assets. These characteristics are more similar to the car rental market, where vehicles that are available at a particular day (or the corresponding available days-of-use) "expire" since they cannot be used in a future time period. Relevant results are obtained regarding dynamic pricing strategies. As this, other important works have dealt with similar environments with insightful outcomes. Adida and Perakis (2010) present an interesting work, where different joint dynamic pricing and inventory control models that deal with demand uncertainty (which depends linearly on price) are considered, within a make-to-stock manufacturing context. This work compares stochastic and robust optimization approaches, introduces different formulations and compares their computational performance.

Nevertheless, the car rental business is characterized by the return of its "sold inventory" in a pre-determined future time period and location. This causes significant changes to the problem structure and renders the problem even more complex to solve. In Oliveira, Carravilla, and Oliveira (2017a), a dynamic programming approach is developed for a deterministic and simpler version of this problem and this question is further discussed.

In innovative transportation systems based on the sharing paradigm, this issue is also present. Bike-sharing, for example, has been a key driver of research on managing capacity to meet demand better, considering variations throughout time and space. Relevant works have focused on capacity reallocation, such as Freund, Henderson, and Shmoys (2017), yet only a few works have focused on the role of pricing in influencing demand, like Chemla, Meunier, Pradeau, Calvo, and Yahiaoui (2013). Nevertheless, this mobility system shows some differences to the car rental business, which hinder the direct application of the developed techniques, such as the homogeneity of the fleet, the design of the repositioning schemes and the motivations (and consequent distribution) of demand, among others. Other innovative businesses are also driving research in this field, such as ride-sharing or e-hailing platforms such as Uber. For example, in Bimpikis, Candogan, and Saban (2016), pricing decisions for a ride-sharing system are introduced to manage the supply-demand balance considering not only variations in time but also the geographical distribution of demand and supply. As the car rental business, the vehicles are shared and thus capacity "returns" to be used by another client. However, the capacity decisions are not centralized in the same decision-maker, as the pricing itself may induce more or less supply of drivers, depending on the business format of the e-hailing platform.

Additionally, the relationship between demand and price in the context of car rental is distinctive and challenging to determine due to the effect of competition and to the myriad of products offered (rental types) that share the same resources (vehicle fleet). Therefore, new approaches are needed to tackle this problem.

1.1.3. Representing uncertainty by scenarios

Scenarios can be important tools for companies dealing with relevant uncertainties. Moreover, the process of scenario generation is critical for the practical relevance of the results obtained.

Scenario generation consists of defining discrete outcomes (realizations) for all random variables and time periods (Høyland & Wallace, 2001), especially useful for stochastic problems. Mitra and Di Domenica (2010) review the scenario generation methods applied in the literature for stochastic programming models, including sampling-based generation (e.g., Monte Carlo, bootstrap or conditional sampling methods), statistical methods (e.g., property matching or regressions) and simulation-based generation (e.g., Vector Auto Regressive methods), as well as other less used methods (e.g., hybrid methods). The authors discuss relevant, desirable characteristics that all scenario generation methods should incorporate: including a variety of factors and existing correlations, considering the purpose of the model (to understand e.g., if it is more relevant to capture variance or higher moments), being consistent with any theory and with empirical data observations. Kaut and Wallace (2003) evaluate different scenario generation methods and propose two properties (and corresponding methodologies to test them) that a method should satisfy to be applicable and relevant to a given problem. Most of these techniques involve a considerable amount of knowledge about the uncertainty and random variables, e.g., their probability distribution.

1.1.4. Methodological approaches

Solving integer stochastic mathematical programming models is becoming a promising approach to obtain good and accurate solutions for complex real-world situations, such as hazard management of post-fire debris flows or transportation network protection against extreme events such as earth-quakes (Krasko & Rebennack, 2017; Lu, Gupte, & Huang, 2018). Often, solution approaches are required to deal with the inherent complexity, such as decomposition or (meta)heuristics (Özcan, 2010; Puga & Tancrez, 2017; Yan, Tang, & Fu, 2008).

Genetic algorithms have been proposed to tackle complex stochastic problems (Gu, Gu, Cao, & Gu, 2010; Wang, Makond, & Liu, 2011). In these works, random variables are often associated with probability distributions; thus scenarios are generated by random sampling or simulation. Furthermore, the hybridization of genetic algorithms and linear programming has been successfully used to develop alternative stochastic methodologies (Reis, Walters, Savic, & Chaudhry, 2005).

In this field, scenario generation is heavily dependent on the knowledge of probability distributions for the random variables and consists on selecting a small set of scenarios that represent it well, which is highly complicated in the multivariate case (Löhndorf, 2016). The author presents an empirical analysis of popular scenario generation methods for stochastic optimization. State-of-the-art methods are compared regarding solution quality, using a problem where analytical solutions are available. Their adequacy is dependent on the problem characteristics and probability distributions. Guastaroba, Mansini, and Speranza (2009) focus on optimal portfolio selection problem and compare scenario generation techniques for this problem. One of the conclusions is that the adequacy of the method depends on the risk profile of the decision-maker.

1.1.5. Core methodological previous works

For this problem, using scenarios to represent uncertainty has a practical interest concerning the application of the method, since scenarios can be useful to help decision-makers understand and act upon the outputs. Nevertheless, the only information regarding the uncertain parameters available for this problem is the bounds on the values they can take. Therefore, a methodology that tackles this lack of information is needed.

Herrmann (1999) proposes a metaheuristic based on genetic algorithms that is especially adequate for problems where the set of scenarios is too large for each element to be evaluated individually, or even known. In this work, the author proposes the co-evolution of solutions and scenarios in two parallel spaces, as follows.

Considering that $\mathcal{SO}$ is the set of all solutions and $\mathcal{SC}$ the set of all possible scenarios, the value obtained by a solution $i \in \mathcal{SO}$ when scenario $j \in \mathcal{SC}$ occurs is given by $F(i, j)$. In this work, the goal was to find the solution that performs best for the worst-case, which is translated (in a minimization problem) to: $\min_{i\in\mathcal{SO}} \max_{j\in\mathcal{SC}} F(i, j)$. The author thus proposes a two-space genetic algorithm where scenarios ($\mathcal{SC}$) and solutions ($\mathcal{SO}$) co-evolve in different populations ($P_{\mathcal{SO}}$ and $P_{\mathcal{SC}}$) composed of individuals whose fitness depends not only on its characteristics but also on the characteristics of the other population. This genetic algorithm favors solutions with better worst-case performances and scenarios with worse "best solutions". The fitness of a solution i_0 is evaluated as $\max_{j\in P_{\mathcal{SC}}} F(i_0, j)$ (worst scenario for this solution), while the fitness of a scenario j_0 is evaluated as $\min_{i\in P_{\mathcal{SO}}} F(i, j_0)$ (best solution for this scenario). The groundbreaking idea in this work is that using efficient genetic algorithms to evolve populations of scenarios requires only an initial sample that will evolve and is thus expected to adequately represent the full set, which would otherwise take significantly more effort to explore. Simultaneously, the solutions evolve to perform increasingly better. This work is continued by other authors, namely Jensen (2001) who proposes a ranking-based evaluation for scenario fitness that performs better and fixes symmetry and bias issues of the original approach.

We aim to extend the idea of a two-space genetic algorithm to evolve solutions and scenarios to other decision-making risk profiles beyond the limitation of the worst-case perspective in Herrmann (1999). Considering the *expected value* as the goal to evaluate solutions (stochastic approach) rather than the worst-case value significantly impacts the evolution of the scenario population. This focuses the evolutionary drive in obtaining a representative population, rather than converging to the worst-case scenario. To achieve this, recent developments on the field of instance generation were considered. In Gao, Nallaperuma, and Neumann (2016), an evolutionary algorithm is proposed for generating instances that are diverse with respect to different features of the problem. It aims to "diversify" points in N-dimensions by ranking candidates based on distance to nearest neighbors in each axis. Using this technique with elitism leads to new children being added to the population only if they extend the extreme values or lie in a large gap between existing points. Also in Deb, Agrawal, Pratap, and Meyarivan (2002), the concept of crowding distance is used to estimate the density of solutions surrounding a particular point in a population. It compares to the largest cuboid enclosing the point without enclosing any other points, with a similar reference to nearest neighbors in each axis.

1.2. Contributions

The main contributions of this paper are related to the mathematical model and the solution methodology proposed.

- We propose a new two-stage stochastic model, extending the deterministic model proposed in Oliveira et al. (2018):
  - Its main innovative feature is that the *stochastic* capacity-pricing problem for car rentals is modeled. Few papers focus on the integration of pricing with capacity decisions, using tactical information and uncertainty to deal with strategic decisions, especially in the complex rental context, where inventory is not depleted but only temporarily unavailable.
  - The issue of vehicle group price hierarchy is included, on a more realistic approach to the problem.
  - Demand uncertainty and price-sensitivity are modeled in an innovative and efficient way, with a significant fit with the problem at hand and its strategic scope. The model is adaptable to different shapes of the demand-price function, considering the effect of competition.
- We propose an innovative solution method to tackle the problem, based on the decomposition of the stochastic model in first-stage and second-stage decisions:
  - Solutions to the first-stage decisions and scenarios are generated in parallel with mutual impact on fitness evaluation, requiring little information on random variables to do so.
  - The fitness depends on the profit obtained by each pair (solution, scenario), which is calculated using a mathematical programming model.
  - The methodology is easily adaptable to different decision-making risk profiles.
  - Specific problem know-how can be used in the initial populations to boost the evolutionary procedure (e.g., providing extreme scenarios).
  - It can be implemented and run in a reasonable time in a decision-support system.

Overall, this methodology has a proper fit with the problem at hand, making it useful in real-world applications. Moreover, it is a

methodology that can be easily extended to other problems where information regarding uncertainty is scarce.

1.3. Paper structure

This paper is structured as follow. Firstly, the problem will be stated and the mathematical model presented (Section 2). Then, Section 3 presents the co-evolutionary matheuristic developed and in Section 4 the results of the computational tests are discussed. Finally, conclusions are drawn and future work and promising research directions are discussed (Section 5).

2. Problem definition

Car rental companies preparing a season must decide on the size and mix of their fleet, i.e., the capacity they will have to face demand in that season. In order for this capacity to be used efficiently, some operational issues that will take place during the season must be considered, as well as the uncertain demand.

A car rental company has several rental stations that share the same fleet. Within the scope of this problem, these stations are often aggregated in regions or locations such that transferring a vehicle between stations within the same region is negligible regarding time or cost, unlike transfers from one region to another. Also, the unit of time considered can be seen as an aggregated measure within this scope, e.g., one week.

The fleet is composed of distinct vehicles, aggregated in vehicle groups that differ in several aspects, namely customer valuation. Nevertheless, the "products" that car rental companies "trade" are *rental types*. Each rental type is characterized not only by the vehicle group requested by the customer but also by start and end time periods and start and end locations (which may be different). Different rental types (products) "compete" for the same fleet (capacity). Moreover, if not conflicting in time, two rental types can use the same vehicle. The demand for each rental type is uncertain and highly price-sensitive. Since it is increasingly easier for customers to compare the prices of all companies offering a certain rental type, brand loyalty plays a relevant role, and if it is not a dominant effect for a certain company, demand is usually mostly attracted by having the lowest price in the market. Due to consumer value perception, the company must also consider constraints on the hierarchy of prices for rental types that are similar in all characteristics except for the vehicle group requested. That is to say, price hierarchy for rentals that start and end at the same time and place must respect the hierarchy of vehicle value, i.e., all other parameters being equal, depending on the vehicle groups considered, a rental price for a more-valued vehicle cannot be less than the price of less-valued one.

Before the season starts, the company must decide on how many vehicles of each group to purchase to meet the (uncertain) demand and where to make them available at the start of the season. Since the pricing strategy heavily influences demand, the company must also decide previously the price it will charge for each rental type. After the season starts and demand is revealed, the company has other tools to meet demand that must be considered since they impact the capacity decisions. On the one hand, since two rentals can use the same vehicle as long as they do not overlap in time, it is critical to decide on fleet deployment throughout the season and network of locations. This deployment is achieved either by actual rentals that start and end in different locations (whose number, limited by demand and capacity, is decided by the company) or by empty transferring vehicles by truck or driver. Pricing is a relevant tool to influence demand and, consequently, fleet deployment and utilization. Also, the company has the possibility to upgrade rentals: offering a more-valued vehicle than requested for the same price. Upgrades are a common practice in this business. Nevertheless, they are used sporadically as a "last resource" to avoid the situation where customers request a less-valued vehicle because they are expecting an upgrade. Finally, to meet temporary peaks in demand, the company may lease more vehicles for a significantly higher cost.

This problem is here modeled as a two-stage stochastic model, where the uncertain parameters are related to demand and competitors' prices. The decisions made before the season starts define the first-stage and, after uncertainty is revealed, the recourse actions or second-stage decisions include the deployment decisions, upgrading, and leasing. The objective of this work is to provide decision-makers with profitable solutions to the first stage decisions, describing their ability to deal with the different realizations of uncertainty, represented by scenarios. The main goal is to obtain good capacity decisions, which will last for a full season. Due to this more strategic setting, aggregated levels of demand and prices are considered. In fact, in this problem, operational decisions such as pricing and its impact on demand are used to get better information for the strategic decisions of capacity, such as fleet size, which cannot be changed throughout the season. Therefore, the problem of updating the pricing policy as the season unrolls – in an "online" manner and considering a fixed capacity – is not considered in the scope of this work.

2.1. Problem modeling

In this section, the uncertain integrated pricing and capacity problem in car rental is fully defined using a mathematical programming model. This model is extended from the deterministic model presented in Oliveira et al. (2018). However, this model differs not only because it considers some parameters to be uncertain but also because it models more accurately the relationships between demand, the price decided and the minimum price in the market. Moreover, it considers that the price charged for a rental requiring a vehicle of a certain group may be limited by the price charged for a rental that only differs on the vehicle group requested (price hierarchy). The notation used is presented in Table 1.

2.1.1. Demand modeling

A generalized relationship between demand, the price decision and the competitors' prices for each rental type is proposed, based on the following assumptions, for each rental type:

1. The demand for a rental type depends on its price, following a given relationship (e.g., a common demand function could follow a linearly inverse relationship with price).
2. If a company has the lowest price in the market, comparing to competitors, it will attract more customers than in the opposite case.
3. For each situation (price above or below minimum competitor price), there is a relationship described by a function dependent on price (see item 1).
4. For each possible price, the demand is higher if the company has the lowest price in the market.
5. The minimum competitor price in the market is an uncertain parameter, within a limited range.
6. The parameters that define the demand functions are unknown (although the shape of the relationship is known).

These assumptions aim to capture and adapt to the scope of this work the price-sensitive and uncertain nature of car rental demand. Fig. 1 graphically represents these assumptions, when the shape of the demand functions is linear. Ainscough, Trocchia, and Gum (2009) present an interesting, although limited, survey on car rental consumers where the effects of rental agency brand and

Table 1
Notation.

Indices, parameters and other notation	
$\theta = \{1, \ldots, \Theta\}$	Index for the set of scenarios
$t, t' = \{0, \ldots, T\}$	Indices for the set $\mathcal{T}$ of time periods[1]
$g, g1, g2 = \{1, \ldots, G\}$	Indices for the set $\mathcal{G}$ of vehicle groups
$s, s1, s2, c = \{1, \ldots, S\}$	Indices for the set $\mathcal{S}$ of rental stations
$r, r' = \{1, \ldots, R\}$	Indices for the set $\mathcal{R}$ of rental types (characterized by check-out station and time period, check-in station and time period, and group requested)
so_r	Check-out station of rental type r
do_r	Check-out time period of rental type r
si_r	Check-in station of rental type r
di_r	Check-in time period of rental type r
gr_r	Vehicle group requested by rental type r
$p_r = \{1, \ldots, P_r\}$	Index for the set $\mathcal{P}_r$ of price levels allowed for rental type r
LBP_r	Lower bound on prices for rentals of type r
UPB_r	Upper bound on prices for rentals of type r
PRI_{rp}	Pecuniary value charged for rental type r related with price level p. The value for $p = 1$ corresponds to the lower bound on price (LBP_r) and for $p = P_r$ corresponds to the upper bound on price (UBP_r). The intermediate levels are a discretization of this range.
$COM_{r\theta}$	Minimum price charged by the competitors for rental type r in scenario θ
$D\tilde{E}M^A_{r\theta}(p)$	Demand function for rental type r in scenario θ, dependent on price level p; valid when price is **a**bove the minimum price in the market for a similar product $COM_{r\theta}$
$D\tilde{E}M^B_{r\theta}(p)$	Demand function for rental type r in scenario θ, dependent on price level p; valid when price is **b**elow the minimum price in the market for a similar product $COM_{r\theta}$, with $D\tilde{E}M^A_{r\theta}(p) \leq D\tilde{E}M^B_{r\theta}(p), \forall p$
MGP	Marginal price difference
PLM_{g1g2}	Whether the price charged for a vehicle of group $g1$ should be lesser than or equal to the price charged for a vehicle of group $g2$, considering the same check-out and check-in locations and time periods, ($= 1$) or not ($= 0$)
COS_g	Buying cost of a vehicle of group g. The value considered is the net cost: purchase gross cost minus salvage value derived from its sale after one year
OWN_g	Ownership cost per time unit of a vehicle of group g
LEA_g	Leasing cost (per time unit) of a vehicle of group g
LP_g	Leasing period for a a vehicle of group g
PYL_g	Penalty charged for each day that a leasing return of group g is late
PYU_{rg}	Penalty charged for each upgrade to vehicle group g applied to rental type r
UPG_{g1g2}	Whether a vehicle of group $g1$ can be upgraded to a vehicle of group $g2$ ($= 1$) or not ($= 0$)
TT_{s1s2}	Transfer time from station $s1$ to station $s2$
TC_{gs1s2}	Transfer cost of a vehicle of group g from station $s1$ to station $s2$
BUD	Total budget for the purchase of vehicles
M	Big-M large enough coefficient
E_θ	Mathematical expectation with respect to scenarios θ
Inputs from previous periods	
Assumption: For all periods, $\mathcal{T}, \mathcal{G}, \mathcal{S}, \mathcal{R}$ are the same, as well as rental types $r \in \mathcal{R}$.	
INX^O_{gs}	Initial number of owned (O) vehicles of group g located at station s, at the beginning of the time period ($t = 0$)
$ONY^{L/O}_{gts}$	Number of owned (O) or leased (L) vehicles of group g on on-going transportation (previously decided), being transferred to station s, arriving at time t
$ONU^{L/O}_{gts}$	Number of owned (O) or leased (L) vehicles of group g on on-going rentals (previously decided), being returned to station s at time t
Other sets	
$\mathcal{R}^g$	Rental types that do not require group g
$\mathcal{R}^{in}_{st}$	Rental types whose check-in is at station s at time t
$\mathcal{R}^{out}_{st}$	Rental types whose check-out is at station s at time t
$\mathcal{R}^{use}_{t}$	Rental types that require a vehicle to be in use at time t

[1] $t = 0$ represents the initial conditions of the time period and "overlaps" with $t = T$ for the previous period.

price are studied. It concluded that the rental agency brand has a positive impact on willingness to rent. These assumptions may thus be suitable for companies with different levels of brand recognition and loyalty, by allowing for different shapes in the demand functions. Furthermore, in the study, the conclusions support the hypothesis that higher prices lead to lower willingness to rent, with no support that they lead to a higher perception of service quality. These conclusions sustain the relatively simple assumptions made within the strategic scope of this model. Lastly, one should notice that different functions may model the demand for each different rental type; nevertheless, the model proposed ensures the interconnection between different vehicle groups arising from possible upgrades and price hierarchy rules.

These assumptions also allow establishing a level of potential exogenous demand. For specific rental types (or all), it might be useful to consider a share of demand that is not influenced by the prices charged by the company. Functions where the value of demand is higher than zero for the entire price domain allow modeling this type of customers.

2.1.2. Mathematical model

Decision variables:

w^O_{gs}	Number of vehicles of group g acquired for the owned fleet available at time $t = 0$ in station s
q_{rp}	$= 1$ if the price charged for rental type r is associated with price level p; $= 0$ otherwise
$w^L_{gts\theta}$	Number of vehicles of group g acquired by leasing at time t to be available at station s in scenario θ
$y^{L/O}_{s1s2gt\theta}$	Number of leased (L) or owned (O) vehicles of group g transferred from station $s1$ to station $s2$ in scenario θ; the transfer begins at t
$u^{L/O}_{rgp\theta}$	Number of rentals of type r that are served by a leased (L) or owned (O) vehicle of group g with a corresponding price of level p in scenario θ
$x^{L/O}_{gts\theta}$	Number of leased (L) or owned (O) vehicles of group g located at station s at the start of time period t in scenario θ
$f^L_{gt\theta}$	Auxiliary variable: total leased fleet of group g at time t in scenario θ
$z_{r\theta}$	Auxiliary variable: $= 1$ if the price charged for rental type r is above the minimum value in the market in scenario θ; $= 0$ otherwise

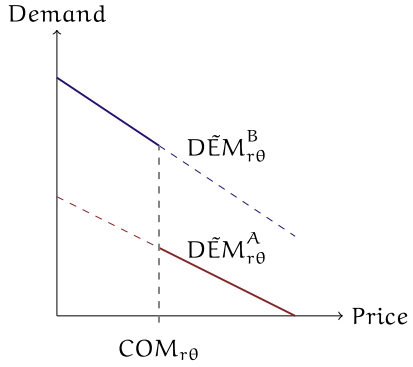


Fig. 1. Relationship between the price decision for a specific rental type r and the minimum competitor price and demand functions, which are dependent on scenario θ.

Optimization model:

$$\begin{aligned}\max \quad & -\sum_{g=1}^{G}\left(\sum_{s=1}^{S} w^{O}_{gs}\right)\left(COS_g + T \times OWN_g\right) \\ & + E_\theta\Bigg[\sum_{r=1}^{R}\sum_{p=1}^{P_r} PRI_{rp}\sum_{g=1}^{G}\left(u^{L}_{rgp\theta}+u^{O}_{rgp\theta}\right) - \sum_{g=1}^{G}\left(\sum_{t=1}^{T} f^{L}_{gt\theta}\right)LEA_g \\ & - \sum_{s1=1}^{S}\sum_{s2=1}^{S}\sum_{g=1}^{G}\left(\sum_{t=1}^{T}\left(y^{L}_{s1s2gt\theta}+y^{O}_{s1s2gt\theta}\right)\right)TC_{gs1s2} \\ & - \sum_{g=1}^{G}\sum_{r\in\mathcal{R}^{g-}}\sum_{p=1}^{P_r}\left(u^{L}_{rgp\theta}+u^{O}_{rgp\theta}\right)PYU_{rg}\Bigg] \end{aligned} \tag{1}$$

$$\textbf{s.t.} \sum_{s=1}^{S}\sum_{g=1}^{G} w^{O}_{gs} COS_g \le BUD \tag{2}$$

$$\sum_{p=1}^{P_r} q_{rp} = 1 \qquad \forall r \tag{3}$$

$$\sum_{p=1}^{P_r} q_{rp} PRI_{rp} \le \sum_{p=1}^{P_r} q_{r'p} PRI_{r'p} \qquad \forall r, r' : \Big\{ so_r = so_{r'} \wedge si_r = si_{r'} \wedge do_r = do_{r'} \wedge di_r = di_{r'} \wedge PLM_{gr_r, gr_{r'}} = 1 \Big\} \tag{4}$$

$$\sum_{g=1}^{G}\left(u^{L}_{rgp\theta}+u^{O}_{rgp\theta}\right) \le q_{rp} D\tilde{E}M^{B}_{r\theta}(p) \qquad \forall r, p, \theta \tag{5}$$

$$\begin{aligned}\sum_{g=1}^{G}\left(u^{L}_{rgp\theta}+u^{O}_{rgp\theta}\right) & \le D\tilde{E}M^{A}_{r\theta}(p) \\ & + \left(D\tilde{E}M^{B}_{r\theta}(p) - D\tilde{E}M^{A}_{r\theta}(p)\right)(1-z_{r\theta}) \qquad \forall r, p, \theta \end{aligned} \tag{6}$$

$$COM_{r\theta} \ge \sum_{p=1}^{P_r} q_{rp} PRI_{rp} - M z_{r\theta} \qquad \forall r, \theta \tag{7}$$

$$\sum_{r\in\mathcal{R}^{out}_{st}}\sum_{p=1}^{P_r} u^{L/O}_{rgp\theta} + \sum_{c=1}^{S} y^{L/O}_{scgt\theta} \le x^{L/O}_{gts\theta} \qquad \forall g, t, s, \theta \tag{8}$$

$$u^{L}_{rgp\theta} + u^{O}_{rgp\theta} \le UPG_{gr_r, g} \times M \qquad \forall r, g, p, \theta \tag{9}$$

$$x^{O}_{g0s\theta} = INX^{O}_{gs} + w^{O}_{gs} \qquad \forall g, s, \theta \tag{10}$$

$$x^{L}_{g0s\theta} = 0 \qquad \forall g, s, \theta \tag{11}$$

$$\begin{aligned} x^{O}_{gts\theta} & = x^{O}_{g,t-1,s,\theta} + ONY^{O}_{gts} + ONU^{O}_{gts} \\ & + \sum_{r\in\mathcal{R}^{in}_{s,t-1}}\sum_{p=1}^{P_r} u^{O}_{rgp\theta} - \sum_{r\in\mathcal{R}^{out}_{s,t-1}}\sum_{p=1}^{P_r} u^{O}_{rgp\theta} \\ & + \sum_{c=1}^{S} y^{O}_{c,s,g,t-TT_{cs}-1,\theta} - \sum_{c=1}^{S} y^{O}_{s,c,g,t-1,\theta} \qquad \forall g, t>0, s, \theta \end{aligned} \tag{12}$$

$$\begin{aligned} x^{L}_{gts\theta} & = x^{L}_{g,t-1,s,\theta} + ONY^{L}_{gts} + ONU^{L}_{gts} \\ & + \sum_{r\in\mathcal{R}^{in}_{s,t-1}}\sum_{p=1}^{P_r} u^{L}_{rgp\theta} - \sum_{r\in\mathcal{R}^{out}_{s,t-1}}\sum_{p=1}^{P_r} u^{L}_{rgp\theta} \\ & + \sum_{c=1}^{S} y^{L}_{c,s,g,t-TT_{cs}-1,\theta} - \sum_{c=1}^{S} y^{L}_{s,c,g,t-1,\theta} \\ & + w^{L}_{gts\theta} \qquad \forall g, 0<t<LP_g, s, \theta \end{aligned} \tag{13}$$

$$\begin{aligned} x^{L}_{gts\theta} & = x^{L}_{g,t-1,s,\theta} + ONY^{L}_{gts} + ONU^{L}_{gts} \\ & + \sum_{r\in\mathcal{R}^{in}_{s,t-1}}\sum_{p=1}^{P_r} u^{L}_{rgp\theta} - \sum_{r\in\mathcal{R}^{out}_{s,t-1}}\sum_{p=1}^{P_r} u^{L}_{rgp\theta} \\ & + \sum_{c=1}^{S} y^{L}_{c,s,g,t-TT_{cs}-1,\theta} - \sum_{c=1}^{S} y^{L}_{s,c,g,t-1,\theta} \\ & + w^{L}_{gts\theta} - w^{L}_{g,t-LP_g,s,\theta} \qquad \forall g, t \ge LP_g, s, \theta \end{aligned} \tag{14}$$

$$\begin{aligned} f^{L}_{gt\theta} & = \sum_{s=1}^{S} x^{L}_{gts\theta} + \sum_{r\in\mathcal{R}^{use}_{t}}\sum_{p=1}^{P_r} u^{L}_{rgp\theta} \\ & + \sum_{s1=1}^{S}\sum_{s2=1}^{S}\sum_{t'=\max\{0,t-TT_{s1s2}\}}^{t-1} y^{L}_{s1,s2,g,t',\theta} \qquad \forall g, t, \theta \end{aligned} \tag{15}$$

$$\begin{aligned} w^{O}_{gs} & \in \mathbb{Z}^{+}_{0} && \forall g, s \\ q_{rp} & \in \{0,1\} && \forall r \\ w^{L}_{gts\theta} & \in \mathbb{Z}^{+}_{0} && \forall g, t, s, \theta \\ y^{L/O}_{s1s2gt\theta} & \in \mathbb{Z}^{+}_{0} && \forall s1, s2, g, t, \theta \\ x^{L/O}_{gts\theta} & \in \mathbb{Z}^{+}_{0} && \forall g, t, s, \theta \\ u^{L/O}_{rgp\theta} & \in \mathbb{Z}^{+}_{0} && \forall r, g, p, \theta \\ f^{L}_{gt\theta} & \in \mathbb{Z}^{+}_{0} && \forall g, t, \theta \\ z_{r\theta} & \in \{0,1\} && \forall r, \theta \end{aligned} \tag{16}$$

The objective function (Eq. (1)) represents the profit obtained by the fulfilled rentals. It considers: the one-time cost of purchasing vehicles and the cost per time period of maintaining this owned fleet, the revenue earned with the rentals fulfilled, and other costs such leasing vehicles, performing empty transfers between stations and an artificial cost to penalize upgrades. The rental revenue is calculated by multiplying the number of rentals fulfilled for a given price level by the pecuniary value associated with that price level. Constraints (5) ensure that the number of

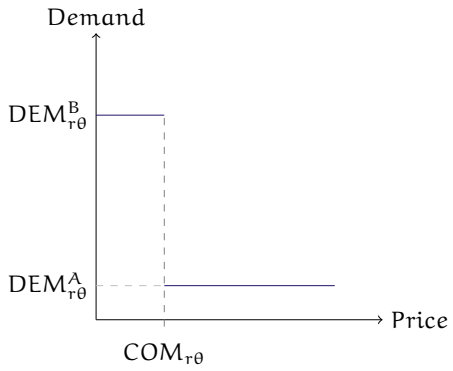


Fig. 2. Relationship between the price decision for a specific rental type r and the minimum competitor price and demand values, which are dependent on scenario θ.

rentals is only positive if the corresponding price level is selected. With this, non-linearity in the objective function is avoided.

Constraint (2) establishes the purchasing budget. Constraints (3) limit the selection of price levels to a single level per rental type (lower and upper bounds on price are guaranteed by the definition of the PRI_{pr} parameters, as explained in the notation section).

A novel issue introduced in this model is the hierarchy among vehicle groups concerning price. Besides being an essential requirement from the business perspective, it introduces some changes to the structure of the problem, relevant for the methodology. More specifically, it is required that the prices for rental types that are similar in everything except vehicle group required follow some hierarchical rules. The goal is to avoid that a luxury vehicle is sold for a smaller price than a compact vehicle, for the same dates and locations. The unitary matrix $PLM_{rr'}$ describes the relationship between groups, indicating whether the price of a group is limited by the price of other. Constraints (4) translate this requirement.

The following constraints have been added or significantly altered compared to the model in Oliveira et al. (2018), based on the assumptions presented and Fig. 1. As mentioned before, Constraints (5) support the modeling of a linear objective function. Constraints (6) limit the number of rentals fulfilled to the existing demand and Constraints (7) relate the price charged for the rental type, the minimum price that the competitors are charging and the demand levels, using binary variables $z_{r\theta}$.

For the remainder of the paper, it was decided to use constant-shaped demand functions, in order to describe a company that is not a market leader operating on a significantly competitive market. That is to say, it is assumed that if the company prices a rental marginally lower than the minimum price in the market for a given scenario ($z_{r\theta} = 0$), it will attract a major slice of the market. If not ($z_{r\theta} = 1$), it will secure only a residual share of the market, as demonstrated in Fig. 2, with the following adaptation of Constraints (5) and (6), where $DEM^{A}_{r\theta}$ and $DEM^{B}_{r\theta}$ are parameters:

$$\sum_{g=1}^{G}\left(u^{L}_{rgp\theta} + u^{O}_{rgp\theta}\right) \leq q_{rp} DEM^{B}_{r\theta} \qquad \forall r, p, \theta \tag{17}$$

$$\sum_{g=1}^{G}\left(u^{L}_{rgp\theta} + u^{O}_{rgp\theta}\right) \leq DEM^{A}_{r\theta} + \left(DEM^{B}_{r\theta} - DEM^{A}_{r\theta}\right)(1 - z_{r\theta}) \quad \forall r, p, \theta \tag{18}$$

Since the rental types share the same resources (vehicles) and there are price hierarchy and substitution issues between groups, such as upgrades, these demand functions still involve a pricing decision and not only a "sell/no-sell" decision, i.e., being above or below the threshold price. Nevertheless, any relationship or shape can be considered for these demand functions. In Section 4.5, a different shape and function will be used to validate this feature.

Constraints (8) limit the vehicles that exit a certain station in a time period by the stock available. Constraints (9) reflect the upgrading policies. Following what is set by the binary parameter $UPG_{gr_r,g}$, they define which vehicle groups requested in the rental type (gr_r) can be upgraded to which vehicle groups (g).

Constraints (10)–(14) define the evolution of the stock variables, as in the previous work. At the beginning of the time period, the stock of owned fleet is given by the initial purchases (10), and there is no stock of leasing fleet (11). For later time periods, in each station and for each rental group, the stock of owned fleet in each scenario is given by the previous stock, increased by the arrival of transfers and rentals from the previous season (parameters) and from previous time periods of the current season and decreased by others that start on this time period (12). For the leasing fleet, the stock is also changed by leasing acquisitions (13) and, when the leasing period expires, removals from the fleet (14). It is assumed that the removal takes place in the same station as the acquisition.

Finally, the auxiliary variables that summarize total leased fleet per group and time period are calculated (Constraints (15)) and the domain of all decisions variables is established (Constraints (16)).

3. Solution method

Two main issues underline the need for a specific solution method to solve the mathematical model presented in the previous section. First, the number of decision variables in a real-world-sized instance is significantly large in this problem. Even in a small case study, a significant amount of rental types (different combinations of starting/ending times and locations) leads to an even higher number of (binary) decision variables, which makes it difficult to solve this model to optimality. However, even if this issue could be overcome for some specific instances with significant computational power, a second issue is related to the need to define and generate valid scenarios for this problem properly. In this work, we propose a methodology that simultaneously generates scenarios and achieves good solutions for the problem. This methodology is based on genetic algorithms, since it is grounded on the previous work by Herrmann (1999) presented in Section 1.

As presented, the methodology proposed in this work is based on the idea of co-evolution between a population of solutions and a population of scenarios to achieve solutions that have a good performance across a diverse set of scenarios. The evolutionary components of the algorithm are based on a biased random-key genetic algorithm (BRKGA) framework (Gonçalves & Resende, 2011), since it is a widely used and well-performing genetic algorithm, which is structured so that the evolutionary procedures are independent of the problem and is available on an API (Toso & Resende, 2015). Appendix A.1, in the Supplementary materials, details the adaptations made to the "problem-independent" part of the original BRKGA framework in order to establish two parallel spaces of evolution.

In this section, the basic co-evolutionary procedures will be discussed, and the problem-dependent parts of the genetic algorithm – decoding chromosomes and calculating fitness – will be detailed for both types of populations.

3.1. Co-evolution of solutions and scenarios

The primary goal of this solution method is to obtain good solutions for the stochastic problem defined in Section 2, for which the only information regarding uncertain parameters are the lower and upper bounds that their values can take. In order to obtain scenarios that have a diverse impact on the solutions, a set of scenarios

will be generated by an evolutionary procedure that is parallel to the procedure of the solutions.

The specific goal of the evolution of the solution population is to obtain values for the *first-stage decisions* of the stochastic model in Section 2.1 that lead to good performance concerning total profit, when compared with the scenario population. The definition of "good performance" depends on the risk profile of the decision-maker and different alternatives will be discussed later in this section. The goal of the evolution of the scenario population is to diversify the impact of its elements on the profit of solutions.

Therefore, the link between these two types of populations is in the calculation of total profit, which involves calculating (or approximating) the *second-stage value function*. For this, the profit resulting from each $<$first-stage solution, scenario$>$ pair is computed. This process will be further discussed in the remainder of Section 3.

3.2. Solution population

3.2.1. Decoder

Solution chromosomes encode solutions to the first-stage decision variables, which must be decided before uncertainty is revealed. These comprehend the purchase of vehicles for the owned fleet and the pricing decisions. When deciding the structure of these chromosomes, it was decided to favor feasibility, i.e., to ensure that the structure always leads to feasible solutions. The first segment of the chromosome refers to the pricing decisions (q_{rp} variables), and it is organized by time period. For each time period t, there is a gene corresponding to each rental type that starts in t. These rental types are ordered so that rental types that require vehicle groups that limit (regarding price) other rental types are decoded first.

Considering the bounds and other limits on prices, some relevant pre-processing steps are applied to the definition of the possible price levels per rental. Let r and r' be two rental types similar in all requirements except for vehicle group and whose vehicle required by r' limits the price of the vehicle group required by r (i.e., $r_{sout} = r'_{sout} \wedge r_{sin} = r'_{sin} \wedge r_{dout} = r'_{dout} \wedge r_{din} = r'_{din} \wedge PLM_{r_g, r'_g} = 1$). Let the parameter *IPL* be the price level interval between levels (e.g. 1 monetary unit). Due to the order of the chromosome, the price for rental type r' is calculated before r.

The maximum value to charge will be given by the smaller value among the upper bound for the specific rental or the price charged for the rental r' that limits it. The minimum value is determined by the lower bound of the rental type. After defining the range of possible prices, the discretization depends on parameter *IPL*. Considering that this range may not be divisible by *IPL*, it was decided that this value will space the first $|\mathcal{P}_r| - 1$ levels, while the latter level $|\mathcal{P}_r|$ will correspond to the higher limit, no matter what the distance is to the previous level.

Therefore, the number of price levels for rental type r is given by:

$$|\mathcal{P}_r| = \left\lceil \frac{\min\left(\sum_{p \in \mathcal{P}_{r'}} q_{r'p} PRI_{r'p}, UBP_r\right) - LBP_r}{IPL} + 1 \right\rceil \tag{19}$$

The values associated with each price level $p \in \mathcal{P}_r$ are:

$$PRI_{rp} = \begin{cases} LBP_r + (p-1)IPL, & p < |\mathcal{P}_r| \\ \min\left(\sum_{p \in \mathcal{P}_{r'}} q_{r'p} PRI_{r'p}, UBP_r\right), & p = |\mathcal{P}_r| \end{cases}$$

This process is exemplified in Fig. 3.

The decoding procedure consists of dividing [0,1[in $|\mathcal{P}_r|$ equal intervals. Based on the value of the gene g_i and in which interval it falls, the price is selected. This fulfills Constraints (3).

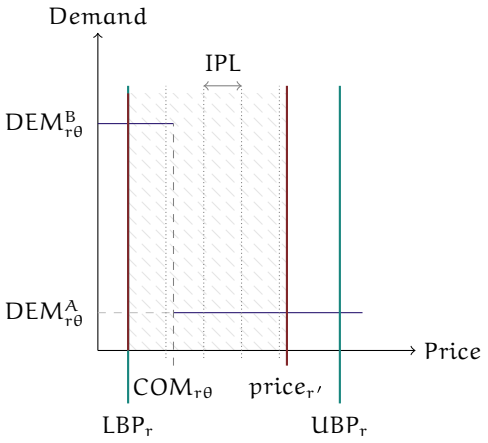


Fig. 3. Definition of price range (hatched area) and corresponding price levels (dotted lines in gray and red lines – the bounds), for a rental type r, whose price is limited by rental type r'. The price decided for r' is here represented as $price_{r'}$ for simplicity. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

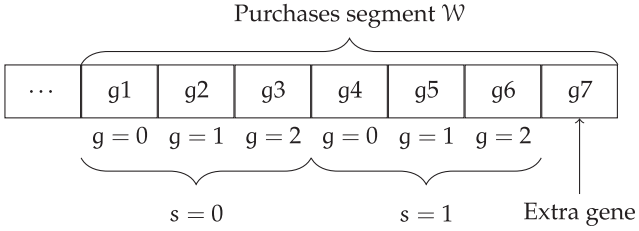


Fig. 4. Structure of a segment of the solution chromosome, corresponding to owned fleet purchase decisions; example for 2 stations and 3 vehicle groups.

$$p = \left\lfloor \frac{g_i}{\frac{1}{|\mathcal{P}_r|}} \right\rfloor \Rightarrow q_{rp} = 1 \tag{20}$$

The following segment of the chromosome, $|\mathcal{S}| \times |\mathcal{G}| + 1$ genes, corresponds to purchase decisions (w^O_{gs} variables). Each gene g_i corresponds to a combination of station and vehicle group, as exemplified in Fig. 4. The fraction of its value over the sum of the values of the chromosomes in the purchases segment $\mathcal{W}$ corresponds to the fraction of the budget that will be assigned to purchase vehicles of this group to be available at this station. The extra gene corresponds to the non-assigned budget. The values are thus given by:

$$w^O_{gs} = \left\lfloor \left(\frac{g_i}{\sum_{j \in \mathcal{W}} g_j} B \right) / \, COS_g \right\rfloor \tag{21}$$

Including the budget as a limit on the purchases incorporates Constraints (2) on the decoding procedure, thus contributing to feasibility goal mentioned above. Nevertheless, it requires these segments of the chromosome to be read twice (once for the calculation of the denominator and once for each numerator), which can lead to a poorer efficiency of the algorithm.

3.2.2. Fitness evaluation

The fitness of an individual determines its ability to survive in a population. For solutions, the goal is to favor those that perform well when faced with the scenario population.

Performance of a solution vs. a specific scenario. As previously introduced, the performance of a specific set of first-stage decisions (a solution) when a specific scenario is revealed is measured by the

profit resulting from solving to optimality (or approximately) the second-stage problem. That is to say, by deciding the best number of rentals to be fulfilled and the best plan for fleet deployment, vehicle leasing and empty transfers – the recourse decisions, which are made after uncertainty is revealed in the form of a scenario – and establishing the resulting profit. By fixing the first-stage decisions encoded in the solution chromosome in the mathematical model presented in Section 2.1, it becomes a smaller model (for the sake of brevity, henceforth designated as second-stage MIP), which is easier to solve. Nevertheless, to speed up the process, an approximation was considered that results from relaxing the integrality constraints on all decision variables of the second-stage MIP, resulting in an LP formulation. The validity of this approximation will be further discussed on Section 4.1.

Performance of a solution across scenarios. Since we lack information regarding the probability distribution of the scenarios, the performance of a solution across all scenarios in the scenario population is computed as the non-weighted average of its performance for each scenario. Nevertheless, decision-makers with different risk profiles value different metrics of performance. Therefore, in order to enrich the information that can be given to the decision-maker, three different decision criteria for solution fitness were established. Consider $\mathcal{SO}$ and $\mathcal{SC}$ to be the set of solutions and scenarios, respectively, within the corresponding populations. Consider $F(i, j)$ to be the profit obtained by solution $i \in \mathcal{SO}$ when scenario $j \in \mathcal{SC}$ is revealed:

- *Laplace criterion*: This is the baseline criterion of expected value in a stochastic approach. As previously explained, due to lack of probability information, the non-weighted average of the total profit obtained for all scenarios is considered as the fitness of a solution i:

$$\text{fitness}_i = \frac{\sum_{j \in \mathcal{SC}} F(i, j)}{|\mathcal{SC}|} \tag{22}$$

- *Pessimist criterion*: Some robust approaches to decision-making favor solutions that perform well when the worst-case scenario is revealed. Since this is a maximization problem, the fitness of a solution according to this criterion is the worst (minimum) profit it obtains across scenarios:

$$\text{fitness}_i = \min_{j \in \mathcal{SC}} F(i, j) \tag{23}$$

- *Optimist criterion*: An optimist approach is also considered, where the fitness of a solution is the best (maximum) profit it obtains across scenarios:

$$\text{fitness}_i = \max_{j \in \mathcal{SC}} F(i, j) \tag{24}$$

3.3. Scenario population

3.3.1. Decoder

Scenario chromosomes encompass information on the uncertain parameters discussed in Section 2: for each rental type, the level of demand if the price is above the minimum in the market ($DEM^A_{r\theta}$), the level if the price is below ($DEM^B_{r\theta}$) and the minimum price of the competitors in the market ($COM_{r\theta}$). In this decoding procedure, it is assumed that the demand-price relationships for each rental type are independent. Due to the aggregate measure of time and space used when defining rental types (e.g., weeks and regions), potential correlations between rentals that start and end in timely and geographical proximity are mitigated. Moreover, the fact that the typical car rental is associated with a previously booked flight journey also weakens the correlation between demand levels for different dates and locations. As Fig. 5 exemplifies, in each scenario

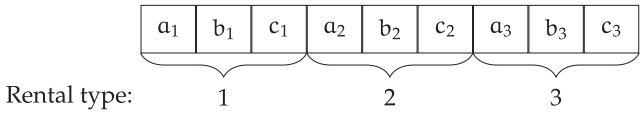


Fig. 5. Structure of a scenario chromosome, for 3 rental types.

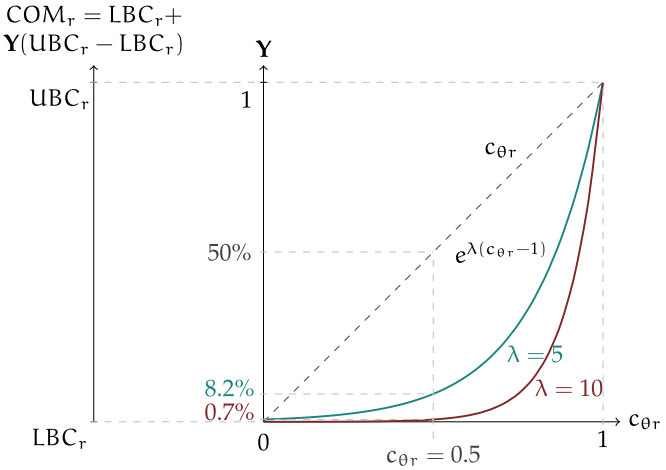


Fig. 6. Exponential relationship between genes of type $c_{\theta r}$ and the uncertain parameter $COM_{r\theta}$ (and its lower and upper bounds LBC_r and UBC_r), exemplified for $\lambda = 5$ and $\lambda = 10$, compared with a linear relationship.

θ, each rental type r is associated with three genes: $a_{\theta r}$, $b_{\theta r}$ and $c_{\theta r}$. Therefore, a scenario chromosome has $3|\mathcal{R}|$ genes. A typical instance of this problem considers 400–2000 rental types, therefore the chromosome size can be a limitation. To decode this chromosome, the following additional inputs are required:

DF_r	Demand forecast for the market of rental type r. It is an upper bound on $DEM^B_{r\theta}$, the demand achieved if the price is the lowest in the market;
$\Delta^{\max}$	Maximum difference between the forecasted demand (DF_r) and $DEM^B_{r\theta}$;
Δ^{A-B}	Maximum difference between $DEM^A_{r\theta}$ and $DEM^B_{r\theta}$;
LBC_r	Lower bound on possible competitor prices;
UBC_r	Upper bound on possible competitor prices;
λ	Parameter to scale the exponential relation between competitor price and the gene value.

The genes of type $b_{\theta r}$ are related with uncertain parameter $DEM^B_{r\theta}$ and define the fraction of $\Delta^{\max}$ that is considered in this scenario. Genes of type $a_{\theta r}$ define a similar fraction for Δ^{A-B}. As for the minimum competitor price $COM_{r\theta}$, it is non-linearly related with genes of type $c_{\theta r}$, as represented in Fig. 6. The parameter $c_{\theta r}$ represents the distance between the minimum competitor price, in the scenario, to its lower bound (considering its full range). As there are several competitors and the minimum prices tend to be consistently closer to their lower bound than to their upper bound, this connection is modeled by an exponential function. In this figure, the impact of using an exponential relationship in the decoding is exemplified, as well as of the choice of parameter λ. For a gene with value $c_{\theta r} = 0.5$, if a "direct translation" were to be used the percentual distance of $COM_{r\theta}$ to its lower bound would be 50% (straight dashed line in Fig. 6). Using an exponential relationship ($\lambda = 5$), this value reduces to 8.2%, and increasing λ will lead to even smaller values. As shown in Fig. 6, with this exponential mapping the probability that the price is closer to the lower bound than to the upper bound is increased.

Summarizing, for each rental type r, based on the genes of the chromosome as presented in Fig. 5, the values of the uncertain parameters are thus obtained:

$$DEM^B_{r\theta} = DF_r(1 - a_{\theta r}\Delta^{\max}) \tag{25}$$

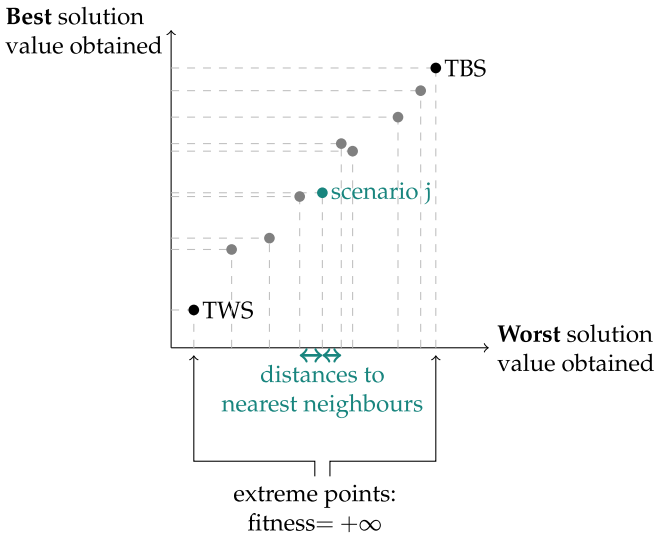


Fig. 7. Example of fitness calculation and distance quantification for the scenario population.

$$DEM^{A}_{r\theta} = DEM^{B}_{r}(1 - b_{\theta r}\Delta^{A-B}) \tag{26}$$

$$COM_{r\theta} = LBC_r + (UBC_r - LBC_r)e^{\lambda(c_{\theta r}-1)} \tag{27}$$

Theoretically extreme cases. In this specific problem, it is possible to define the theoretically extreme cases. In theory, the best scenario is the one where demand is always the highest possible and competitor prices are the lowest (and vice-versa for the worst scenario). These two extreme scenarios, TBS (Theoretically Best Scenario) and TWS (Theoretically Worst Scenario) are included in the first generation of the scenario population, which is otherwise randomly generated. When using this method with other real-world problems, if some scenarios but not the totality of them are known, they can also be added to the initial scenario generation. Also, if the extreme cases are not previously known, the evolution of the scenario population is expected to converge to include those values.

3.3.2. Fitness

Having established the goal of obtaining a diverse and representative population of scenarios, the fitness evaluation must ensure that individuals that contribute the most for this goal survive in the population. *Diversity* is considered regarding the impact that the scenarios have on the profit of the solutions. A diverse scenario population consists of scenarios that result in different profits for the same set of solutions.

The fitness of an individual scenario translates its contribution to the population diversity and is based on the distance to other scenarios, regarding the difference in total profit obtained by the solutions. The methodology to compute distance is based on research in feature-based diversity optimization for instance generation discussed in Section 1.

Each scenario j is mapped on a bi-dimensional space, according to two correlated features: the best value obtained by a solution when it is unveiled $\max_{i\in\mathcal{SO}} F(i,j)$ and the worst value obtained $\min_{i\in\mathcal{SO}} F(i,j)$. Fig. 7 exemplifies this procedure. For each feature, or axis, the scenarios that represent extremes are given a very high fitness value, in order to favor scenarios that broaden the "space" occupied by population. For the remaining scenarios, the nearest neighbors are identified and the product between the distances to each of the neighbors is computed. The fitness of a scenario is thus the maximum value between the product of distances in both axes. With this, scenarios that "fill in gaps" within the space the population occupies are favored. Algorithm 1 details the steps of this calculation.

Algorithm 1 Pseudocode for scenario fitness calculation

Require: Matrix $|\mathcal{SO}| \times |\mathcal{SC}|$ of profit values $F(i,j)$
  tuple $B(id, best) \leftarrow \emptyset$
  tuple $W(id, worst) \leftarrow \emptyset$
  for all $j \in \mathcal{SC}$ **do**
    $B_j \leftarrow (j, \max_{i\in\mathcal{SO}} F(i,j))$
    $W_j \leftarrow (j, \min_{i\in\mathcal{SO}} F(i,j))$
  end for
  sort B in ascending order
  sort W in ascending order
  $Bdist \leftarrow \emptyset$
  $Wdist \leftarrow \emptyset$
  for all $j \in \{2, \ldots, |\mathcal{SC}| - 1\}$ **do**
    $Bdist_{B_j.id} \leftarrow (B_{j+1}.best - B_j.best) \times (B_j.best - B_{j-1}.best)$
    $Wdist_{W_j.id} \leftarrow (W_{j+1}.worst - B_j.worst) \times (B_j.worst - B_{j-1}.worst)$
  end for
  fitness $\leftarrow \emptyset$
  for all $j \in \mathcal{SC}$ **do**
    if $j = B_1.id$ OR $j = B_{|\mathcal{SC}|}.id$ OR $j = W_1.id$ OR $j = W_{|\mathcal{SC}|}.id$ **then** $\text{fitness}_j \leftarrow +\infty$
    else $\text{fitness}_j \leftarrow \max(Bdist_j, Wdist_j)$
    end if
  end for
  return fitness

In this case, evaluating the fitness of a scenario j' implies knowing not only the value of $F(i,j')$ for all $i \in \mathcal{SO}$ but also the F value of the other scenarios $j \in \mathcal{SC}$ for all $i \in \mathcal{SO}$. This highlights the relevance of firstly computing the matrix $F(i,j)$ for each combination of solution and scenario and afterwards calculate the fitness values (see Fig. 14(b) in Appendix A.1), so as to apply Eq. (1) only once per pair (i,j).

4. Computational experiments, results and discussion

The goal of the computational experiments discussed in this section is to validate the value of the methodology proposed in terms of: (i) the quality of the solutions proposed, (ii) the diversity and representativeness of the generated set of scenarios, which support the "robustness" of the solutions, and (iii) the applicability and utility of the method when integrated in a decision-support system. Additionally, to validate the simple adaptation needed to use a different shape for the demand function, some tests with linear functions are run and the results compared.

Instances. To test the methodology proposed, a set of instances were adapted from the set of realistic instances for the (deterministic) car rental capacity-pricing problem, available in Oliveira, Carravilla, and Oliveira (2017b). These instances are based on real data provided by a car rental company and the adaptation procedure, detailed in Appendix A.2 in the Supplementary materials, maintains the key characteristics and size of the information regarding fleet and rentals.

In the intensive computational tests, the instances are run with ten different seeds for each of the three different solution fitness criteria. The first eight instances from the original set were used for the comprehensive tests (30 runs per instances). Six other instances (selected to represent different sizes) were run once (for

Table 2
Main characteristics of the instances.

Instance	Size indicator $(\lvert\mathcal{R}\rvert \times \lvert\mathcal{G}\rvert)$	Market size factor	# runs
1	428	Small	30
2	428	Large	30
3	486	Small	30
4	486	Large	30
5	517	Small	30
6	517	Large	30
7	1124	Small	30
8	1124	Large	30
17	2772	Small	1
18	2772	Large	1
29	4184	Small	1
30	4184	Large	1
37	6170	Small	1
38	6170	Large	1

one seed and one fitness direction criterion) in order to draw some conclusions regarding computational time. This will be further discussed later in this section. Table 2 presents the main characteristics of the instances used. The size of the instance is approximated by the number of rental types and vehicle groups it considers. For the same size indicator, two different market size factors are considered: small and large. The difference between these two instances is that the reference values for the demand, as well as the budget, are in a *large market* 100 times larger than in a *small market*. As will be discussed throughout this section, this has a significant impact on the complexity of the instances.

Parameters. The stopping criterion for the genetic algorithm is the maximum number of generations and was set to 3000. The number of chromosomes in the solution and scenario populations were set to 20 after preliminary tests showed that these values allowed for the algorithm to perform well. The genetic algorithm was based on the brkgaAPI released (see Appendix A.1 for the specific alterations) and the remaining parameters were set to match the proposed default parameters (Gonçalves & Resende, 2011; Toso & Resende, 2015).

Technical details. The tests were run on a server Intel(R) Xeon(R) X5690 with 3.46 gigahertz (2 processors), and 48 gigabytes RAM. The MIP and LP solver used was CPLEX 12.6.3 and the algorithm was coded in C++.

4.1. Preliminary tests: validating the LP approximation on fitness calculations

In order to speed up the solution method proposed, an approximation was considered for calculating the best profit that a specific solution can achieve if a certain scenario is realized. This approximation consists of relaxing the integrality constraints on all decision variables of the second-stage MIP that provides the optimum profit value required, resulting in a linear program (LP). The goal of these preliminary tests is to validate this approximation.

The most relevant impact of using an approximation is not on the profit values obtained per se but on the differences caused on the rank of the solutions and scenarios according to their fitness. Therefore, to validate this approximation, the fitness of the last generation of solutions and scenarios was calculated based on the LP and the second-stage MIP, for each run of the first eight instances. The consequent order of the solutions and scenarios was compared for both cases. The order according to which the individuals were ranked was the same using the two approaches, for both solution and scenario populations. Moreover, the differences in fitness value were negligible, with no significant differences between solution fitness criteria. Table 12 in Appendix A.3, in the Supplementary materials, presents the detailed results. This validates the approximation considered in this solution method.

4.2. Solution evolution

The goal of the tests presented in this subsection is to assess the ability of the method to generate good solutions. For this, we study the solution fitness improvement over generations and overall run times. To further understand the quality of these solutions, approximations of the Expected Value of Perfect Information (EVPI) and the Value of the Stochastic Solution (VSS) are analyzed.

Table 3 presents the main results associated with the evolution of the solution populations, as well as the overall computing times. The fitness of the solutions is assessed according to the different criteria presented in Section 3.2, as well as the final best value. As mentioned before, the fitness value is computed by solving the LP approximation of the second-stage MIP, while the final best values presented result from solving the MIP models of the last generation of the genetic algorithm to optimality. It is possible to observe that there is a higher standard deviation in the results obtained when the Laplace criterion is used. This is primarily due to the impact of the different scenario populations generated in each run. This effect will be further discussed when comparing the evolution of solution fitness throughout generations.

The percent improvement on solution fitness throughout the genetic algorithm is also presented. It is possible to observe that this metric is influenced by the fitness criterion selected. For runs where the Pessimist criterion guided the solution evolution, this improvement is significantly larger in average (380% vs. 62% and 59%). This effect is consistently present in all instances. This difference can be explained by the characteristics of the problem, namely the impact on profit of fleet utilization levels, linked with the random construction of the initial generation of solutions. If demand and market prices are high (in the best scenarios), random solutions, at worst, lead to a fleet smaller than what it should be. Nevertheless, there is a high level of utilization of the small fleet capacity and all vehicles tend to generate profit. However, if demand and market prices are low (in the worst scenarios), similar random solutions will overestimate fleet capacity, which will lead to lower utilization of the fleet and consequently higher costs for lower revenues. Therefore, the room for improvement is larger when considering the performance on worse scenarios.

Fig. 8 shows an example of solution fitness evolution throughout the genetic algorithm. Three runs with different solution fitness criteria are compared. As expected, the scale of the fitness values is significantly different for each criterion. Nevertheless, it is possible to observe the evolution and convergence of the fitness values. For Laplace criterion, despite significant oscillations, there is a convergence around an average value. These oscillations are related to changes in the scenario population, which is also evolving. For Pessimist and Optimist criteria, the evolution profile is not oscillating since the extreme scenarios (TWS and TBS) are known for this problem, and were included in the initial generation of the scenario population (see Section 3.3).

Table 3 also quantifies the average computational times. It is possible to see they are influenced by instance size and market size factor. Despite this increase in runtime, it is important to notice that, due to the strategic scope of the decisions here considered, these values are reasonable for the application of this methodology as a decision-support tool. This issue will be further discussed on Section 4.4.

Finally, to further validate the performance of this method as far as the quality of the solutions generated is concerned, the results obtained for a fixed scenario population were compared with the optimal solutions for the three different fitness criteria. A

Table 3
Overall results of best solution fitness and final value and computational times.

Instance	Solution fitness criteria	Improvement on solution fitness (last vs. initial generation)	Final best value (according to fitness criterion)	Std deviation of final best value	Time (seconds)
1	Laplace	76%	37,728.8	13%	1,288
	Pessimist	1026%	5,777.5	3%	1,407
	Optimist	54%	80,555.2	0%	1,310
2	Laplace	24%	4,037,570.0	8%	2,450
	Pessimist	83%	1,001,230.0	1%	2,413
	Optimist	54%	8,067,420.0	0%	2,451
3	Laplace	59%	45,993.1	16%	1,458
	Pessimist	441%	8,039.4	3%	1,593
	Optimist	51%	98,043.0	0%	1,493
4	Laplace	25%	4,933,750.0	13%	2,950
	Pessimist	76%	1,309,700.0	1%	2,928
	Optimist	55%	9,797,250.0	0%	2,914
5	Laplace	54%	55,429.3	24%	1,641
	Pessimist	239%	10,395.6	3%	1,787
	Optimist	51%	119,184.0	0%	1,673
6	Laplace	25%	6,071,930.0	14%	3,155
	Pessimist	70%	1,681,320.0	1%	3,110
	Optimist	52%	11,925,700.0	0%	3,150
7	Laplace	187%	32,385.5	19%	8,915
	Pessimist	991%	3,965.8	7%	9,353
	Optimist	79%	67,439.3	1%	9,018
8	Laplace	44%	3,476,000.0	14%	16,191
	Pessimist	116%	778,926.0	1%	16,195
	Optimist	75%	6,784,230.0	1%	16,238
Laplace average: 62%					
Pessimist average: 380%					
Optimist average: 59%					

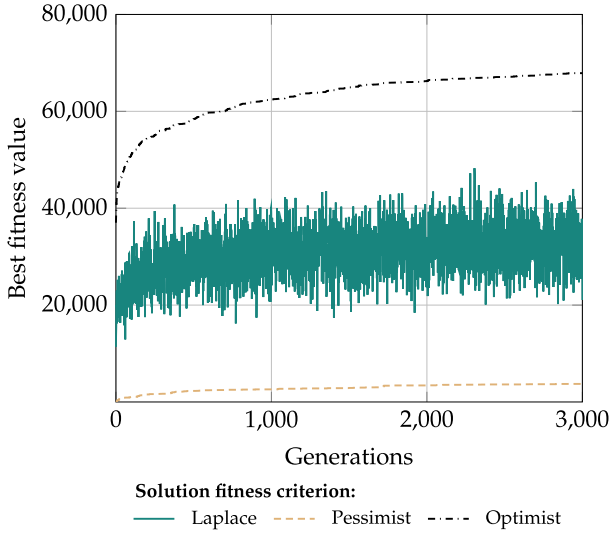


Fig. 8. Evolution of solution fitness, for the different solution fitness criteria (example from instance 7).

smaller instance was developed to enable CPLEX to achieve the optimal solution within a reasonable time. This instance was adapted from Instance 1 and is similar to it in all aspects except for the rental types considered, resulting in a size indicator of 100 (confront with Table 2). The same fixed set of scenarios was considered both for the solver and for the heuristic runs. For each fitness criterion, the solver reached the optimal solution, and the proposed solution method was run with ten different seeds, resulting in an average final best value less than 1% below the optimum. More specifically, the difference between the average final best value and the optimum was of −0.2%, −0.4%, and −0.6% for Laplace, Pessimist, and Optimist criteria, respectively. Even though one of the main advantages of the matheuristic proposed is the simultaneous generation of scenarios, which is not assessed in this experiment, this result supports the assertion that the solution method is able to achieve good solutions for this problem.

4.2.1. Expected value of perfect information (EVPI) and value of the stochastic solution (VSS)

The solution method is a matheuristic procedure and thus the results it obtains – concerning solutions and scenarios – are not proven to be optimal. Nevertheless, interesting insights can be drawn by developing an approximated measure of the Expected Value of Perfect Information (EVPI) and the Value of the Stochastic Solution (VSS), assuming the set of scenarios generated is representative. In this context, the terms EVPI and VSS, as well as other terminology from Stochastic Programming such as recourse problem value or wait-and-see solutions, will be loosely used for simplicity, representing an approximation derived from a non-exact methodology. Fig. 9 presents the framework for obtaining these values. The goal is to obtain an approximation of how important it is to have good forecasts in this problem (EVPI) and of how much a company can profit from applying this stochastic method instead of a deterministic one (VSS).

EVPI is the difference between the best value obtained by the stochastic method or recourse problem (RP) and the wait-and-see value (WS). We use the Laplace criterion for solution fitness when computing the RP throughout this section, in order to be comparable with the other values that are also results of the (non-weighted) average of scenarios. The WS is the average for all scenarios of the profit values obtained if the decisions

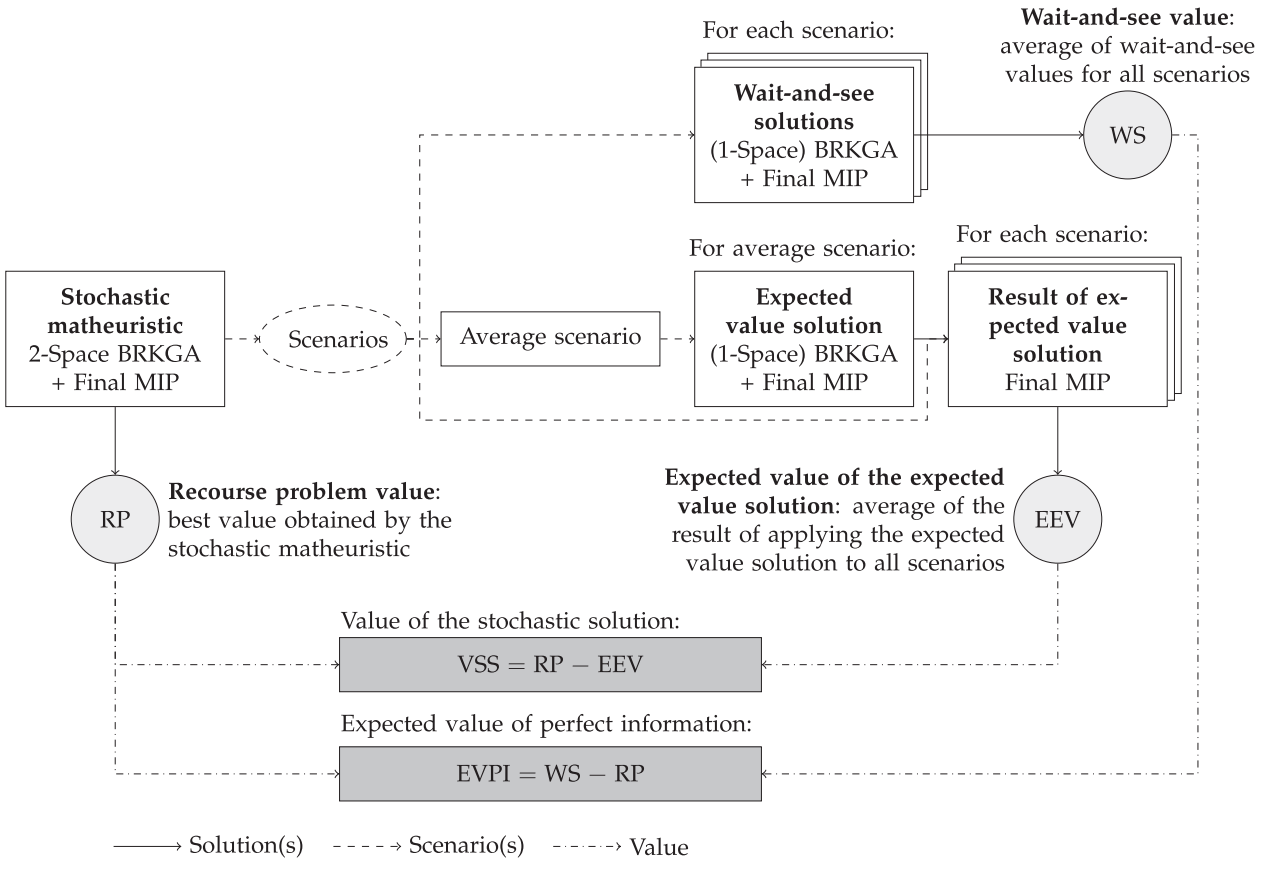


Fig. 9. Framework for calculating VSS and EVPI, considering the Laplace criterion for solution fitness (for a maximization problem).

Table 4
Average measures of Expected Value of Perfect Information (EVPI) and Value of the Stochastic Solution (VSS), using indicators of Recourse Problem value (RP), Wait-and-See value (WS) and Expectation of using Expected Value solution (EEV).

Instance	RP	WS	EEV	EVPI = WS−RP	EVPI% EVPI/RP	VSS = RP−EEV	VSS% VSS/EEV
1	37,729	40,991	30,506	3,263	9%	7,223	24%
2	4,037,570	4,244,760	3,730,540	207,190	5%	307,030	8%
3	45,993	49,663	37,902	3,670	8%	8,091	21%
4	4,933,750	5,183,560	4,530,280	249,810	5%	403,470	9%
5	55,429	59,697	45,522	4,268	8%	9,907	22%
6	6,071,930	6,335,700	5,506,880	263,770	4%	565,050	10%

were made knowing the scenario in advance (perfect information). The stochastic method is a *matheuristic* that consists on evolving a Two-Space BRKGA where solutions and scenarios co-evolve and, for the last generation populations, solving MIP models to calculate the exact impact of each scenario on each solution. In order to obtain a comparable value, a similar matheuristic was run to generate wait-and-see solutions. The difference is that this One-Space BRKGA evolves only the solution population, considering a fixed scenario (deterministic).

VSS compares the best value obtained by the stochastic method (RP) with the expected result across scenarios of applying a deterministic solution that results from considering only an average scenario (EEV). A similar One-Space BRKGA was also used to generate the expected value solution. The impact of this solution across scenarios was computed by solving MIP models.

For this computation, the first six instances, which are similar in size, were used. Table 4 presents the results obtained.

Regarding the EVPI, it is possible to conclude that, assuming the set of scenarios is representative and considering that genetic algorithms as described above are used to make decisions, improvements between 4% and 9% can be expected if the uncertainty is removed. This value is helpful for companies to understand how much to invest in better forecasting methods, for example.

As for the VSS, it measures more directly the impact of using this stochastic method instead of a similar deterministic one. The instances' market size factor significantly impacts these values. For smaller market sizes (instances 1, 3 and 5), considering uncertainty results on 21–24% improvements on expected profit. Large market instances (2, 4 and 6) show less significant results, with nonetheless significant improvements of 8–10%.

4.3. Scenario evolution

The goal of the tests presented in this subsection is to assess the quality of the final scenario population obtained by the proposed method. In these computational experiments, each population consists of 20 scenarios. For this, two main characteristics are studied: diversity – how different the scenarios in the population are, and representativeness – how well the set of scenarios represents the possible ranges of impact on the solutions.

As mentioned when discussing the results of solution fitness evolution on Section 4.2, the oscillation on solution fitness when

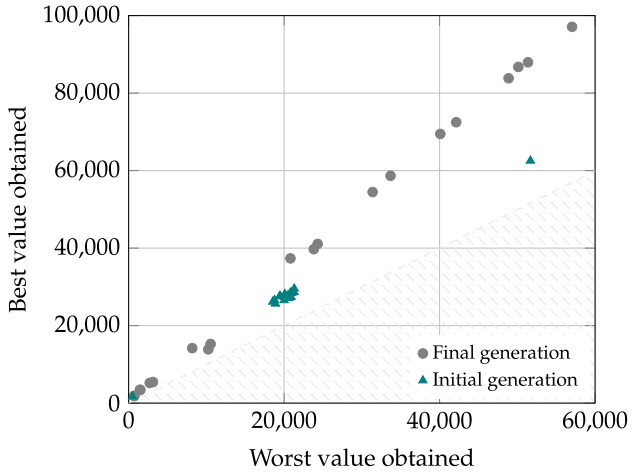


Fig. 10. Representation of the scenarios in the initial and final generation of the scenario population (example from instance 3, run with Laplace solution fitness criterion). The two axes refer to the best and the worst values achieved by the solutions in the solution population for each scenario. Considering the notation introduced in Section 3.2, for a scenario j: $\max_{i \in \mathcal{SO}} F(i, j)$ and $\min_{i \in \mathcal{SO}} F(i, j)$, correspondingly. The hatched area represents an *a fortiori* impossible region.

using the Laplace criterion (see Fig. 8) is caused by the non-monotonous evolution of the scenario population. In fact, due to the strategy for increasing diversity on the scenario population, scenarios are ranked regarding how different is the impact they have on the solutions. This means that a single individual entering or leaving the population can cause the fitness of another individual to change drastically. However, in this work, quantifying the value or fitness of each scenario is a means to evolve towards a population with certain characteristics. The focus is not the individual fitness of scenarios but on the structure of the population as a whole. Two main characteristics of the scenario population are considered:

- Diversity, since the goal is to obtain scenarios that are different concerning the impact they have on the solutions, and
- Representativeness, since the scenario population does not comprise all possible scenarios and we need to ensure that the performance of the solutions against these is indicative of their performance against all possible scenarios.

4.3.1. Diversity

The diversity of a scenario population is connected with the scenario fitness calculation presented in Section 3.3. Fig. 10 is based on the bi-dimensional system previously introduced (exemplified in Fig. 7), where each scenario in a population is mapped according to the worst and the best values it "causes" on the solution population. Fig. 10 compares the initial and last generation of the scenario population. The initial population (green triangles) has two clear extreme points (TWS and TBS – see Section 3.3) and the remaining scenarios, which were randomly generated, are close to each other. The last generation (gray circles) is spread between these two extremes, showing the effect of evolution. The hatched area represents an *a fortiori* impossible region (the worst value cannot be better than the best value obtained). The fact the population is not spread across the remaining space is due to the fact that the two features that are used to map scenarios are correlated, i.e., the higher the worst value is, the higher the best value tends to be. Especially considering that the solution population converges to similar and well-performing solutions, in the last generation it is not expected that a single scenario is simultaneously adverse for one solution and favorable for other. Based on this mapping, a possible measure of population diversity is the average Euclidean distance between the coordinates of the scenarios projected in this 2-axes system. There is an average fourfold increase in this metric when comparing initial and final populations for all runs. Table 13 in Appendix A.4, in the Supplementary materials, details these results.

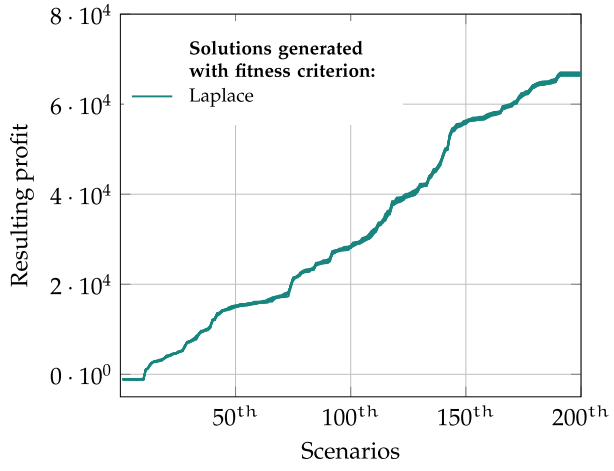
Table 5
Difference in average profit obtained by the best solution when facing all scenarios generated in different runs (of the same solution fitness direction) vs. facing co-generated scenarios.

	Percent increase in average profit		
	Solution fitness criteria		
Instance	Laplace	Pessimist	Optimist
1	1.6%	0.7%	1.3%
2	0.6%	0.2%	0.5%
3	2.6%	1.0%	6.0%
4	1.9%	0.5%	1.0%
5	6.0%	3.5%	2.9%
6	1.8%	1.7%	2.0%
7	4.1%	4.1%	1.9%
8	2.2%	0.7%	3.3%

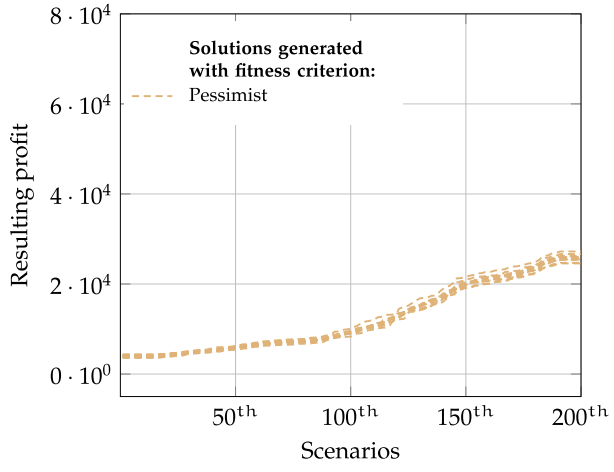
4.3.2. Representativeness

Since there is no better information available regarding uncertainty than the bounds used in the scenario decoder, it is difficult to evaluate the representativeness of scenario populations concerning the *accuracy* of the generation method. Nevertheless, it is possible to evaluate it in terms of its *precision*. Representative scenario populations should have a similar impact on the solutions. To test this, the best solutions found for each instance were evaluated against the last generation of 20 scenarios from the same run and against the scenarios generated in different runs (for the same instance and solution fitness criterion). This last set of scenarios is composed of 200 elements, corresponding to the last generations of 20 scenarios for each of the 10 runs. The average values of the profit obtained by solving the second-stage MIP problems across the two different sets of scenarios were compared. Table 5 presents the results per instance and solution fitness direction. On average, the impact is similar, ranging from 0% to 6%. Nevertheless, there is a slight tendency of the co-generated scenarios to underestimate profit obtained. Although the average values in Table 5 are all positive, for some runs/seeds the difference is negative. Therefore, the fact that a single run of the method results in scenarios that slightly underestimate profit for the best solution is an average trend rather than a general output.

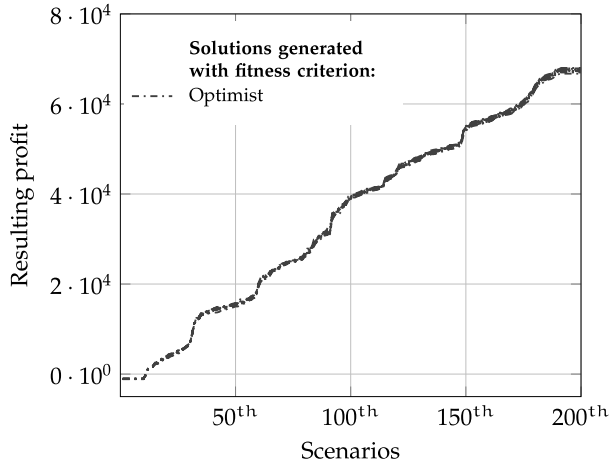
Besides analyzing the precision of the scenario population regarding the average impact on profit, it is important to evaluate this precision on its *impact across scenarios*. For this, the solution fitness criteria play an important role, since the criterion used significantly influences the structure of the final solutions generated. For the Pessimist and Optimist criteria, the performance across all scenarios is not considered at any point of the evolutionary procedure and it becomes an essential point of study in this section. When considering the problem at hand, a relevant first-stage decision that is critical for this performance is the fleet size. If the decision is made solely considering the worst-case scenario (Pessimist criterion) – where demand is lower – the company will tend to acquire a smaller fleet so that it does not incur in costs with the non-utilized fleet. Therefore, if the demand increases considerably (best-case scenarios), the company may not be able to seize all the possible revenue due to the lack of vehicles. In the opposite case, Optimist decisions lead the higher fleet size decisions, causing unnecessary costs in scenarios where demand is lower.



(a) Laplace criterion



(b) Pessimist criterion



(c) Optimist criterion

Fig. 11. Resulting profit of applying the best solutions generated in each run against the full set of scenarios that were generated considering the same solution fitness criterion (instance 7 – small market).

Fig. 11 presents how the best solutions generated in instance 7 perform across all scenarios generated for the same solution fitness criterion. The conclusions drawn are similar for all instances.

The solutions perform similarly across all scenarios when the solution fitness criterion is the same. This supports the representativeness of the scenario populations generated by this method concerning the impact they have on the "robustness" of solutions. That is to say, a solution co-generated with a small set of scenarios has the same behavior than a solution co-generated with another set of scenarios. This effect is more visible on solutions generated by the Laplace and Optimist criteria (for simplicity, henceforward named as "Laplace solutions" and "Optimist solutions"). For those generated by the Pessimist criterion ("Pessimist solutions"), as the scenarios get closer to the best case, differences between solutions become more evident. Since the solution evaluation depends only on the performance in the worst case, it is expected that the evolution of solutions leads to different performances on the best cases.

There is also an observable difference between the performance profile of Pessimist solutions when compared to those of Optimist or Laplace solutions, which is related to the goal of each evolution strategy. Pessimist solutions perform better than Laplace and Optimist solutions in the worst cases yet not in the best cases or on average. This observation is coherent with the previous discussion on the solution fitness criteria impact on fleet size decisions. In fact, throughout all best solutions retrieved from all instances, there is a relevant difference between the non-used budget on Pessimist solutions (21%) and Laplace and Optimist solutions (both 1%). For Laplace solutions, the budget is expected to be used similarly to Optimist solutions, since the best scenarios only compensate the bad in average if the company has enough cars to sell when demand increases.

Moreover, the similarity between Laplace and Optimist solutions is visible not only on performance but in part of the solution structure as well. Fig. 12 shows how the solutions in Fig. 11 translate into fleet purchase decisions. Besides performing similarly across scenarios, Laplace solutions have a similar structure among themselves regarding fleet size and mix decisions, as well as Optimist solutions. Also, there is some similarity between the structure of solutions generated by these two criteria. Pessimist solutions, however, have a distinct structure from Laplace and Optimist solutions (concerning the vehicles purchased for Location 1, for example). Moreover, they show larger differences among themselves, including total fleet size values. These differences are, as discussed, a clear indicator of performance dissimilarity in scenarios with higher demand.

4.4. Decision support

Ultimately, the goal of this methodology is to support decision-makers, by providing a set of good solutions that are appropriate for different risk profiles and by helping the visualization of the different impact uncertainty can have on these solutions. In this section, the outputs of the method that are relevant for decision-makers are discussed, as well as possible computational limitations.

The number of solutions that the decision-maker can obtain with this method depends on the time or computational resources available. Even if these are scarce, with only one run per solution fitness criterion, three different and good solutions can be compared. As an example, Fig. 13 shows the best results achieved in these conditions for instance 8. The vertical axis represents the resulting profit from each of the best solutions if each scenario occurs. Following the direction of improvement of scenarios, vertical lines show points where the lead among the three best solutions changes. For the worst scenarios, the Pessimist solution performs slightly better than the others. Then, there are a few scenarios where the resulting profit is similar among the three solutions. These scenarios are followed by a significant portion of scenarios where the Laplace solution outperforms the others. Finally, for the best scenarios, the Optimist solution has the best results, closely followed by the Laplace solution, while the Pessimist solution falls behind.

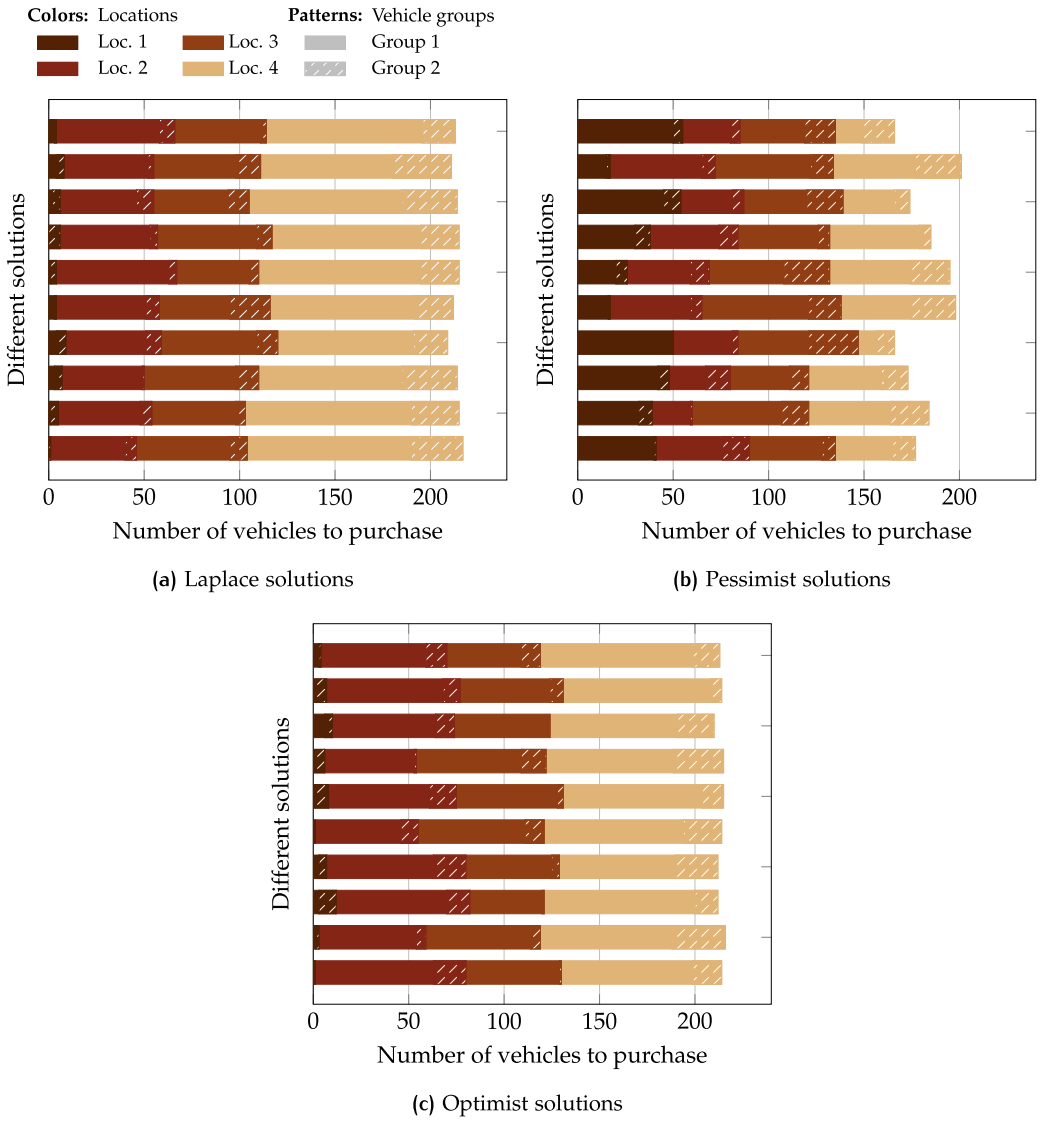


Fig. 12. Fleet size/mix decisions for the solutions presented in Fig. 11: number of vehicles to purchase per vehicle group and available location.

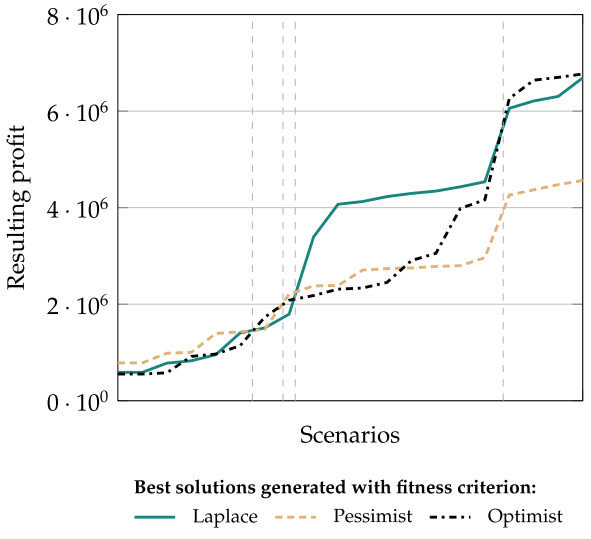


Fig. 13. Final best solutions generated (example from instance 8).

In order to make a decision, it is important to understand what differentiates the structure of these solutions. Table 6 presents the main characteristics of the three solutions exemplified in Fig. 13, concerning the capacity decisions. It is possible to see that the Pessimist solution has a higher percentage of the budget that is not used for purchases, leading to a smaller fleet size. This partly explains why this solution is not able to perform as well in scenarios with high demands. Other structural insights are related to the vehicle groups. It is possible to see that all solutions favor Group 1 (the less-valued vehicle group), yet this effect is magnified for Optimist and Laplace solutions. As for the rental locations where the purchased vehicles are made available, in this case, there are only slight differences between solutions. Nevertheless, this metric can be relevant for the decision-maker in other situations.

The tools developed in this methodology can be applied in interesting features of a decision-support system. For example, it would be possible for a decision-maker to change the solutions found and test these new solutions against the pool of scenarios that were generated in the process. Also, the decision-maker can feed some scenarios to the initial population. Nevertheless, some limitations must apply, namely the need to maintain a minimum number of randomly generated individuals on the population.

Table 6
Characteristics of fleet capacity spending for the solutions presented in Fig. 13.

		Solutions in Fig. 13		
		Laplace	Pessimist	Optimist
Non-allocated budget		1%	30%	1%
Weight of vehicle group in purchases	Group 1	74%	68%	79%
	Group 2	26%	32%	21%
Weight of locations as destinations of purchased vehicles	Location 1	30%	28%	25%
	Location 2	32%	28%	29%
	Location 3	19%	17%	20%
	Location 4	20%	27%	26%

Table 7
Average run times.

Instance	Size indicator $(\lvert\mathcal{R}\rvert \times \lvert\mathcal{G}\rvert)$	Market size factor	Average time (seconds)
1	428	Small	1,335
2	428	Large	2,438
3	486	Small	1,515
4	486	Large	2,931
5	517	Small	1,700
6	517	Large	3,138
7	1,124	Small	9,096
8	1,124	Large	16,208
17	2,772	Small	26,982*
18	2,772	Large	69,480*
29	4,184	Small	32,607*
30	4,184	Large	43,021*
37	6,170	Small	46,072*
38	6,170	Large	43,591*

*Based on a single run.

A possible drawback of the applicability of this methodology is the significant increase of computational effort required if the size of the instances increase. As mentioned at the beginning of Section 4, the intensive computational tests performed required instances to be run with ten different seeds for each of the three different solution fitness criteria. All instances available are realistic yet different in size. In order to assess the computational time required for larger instances, six instances with larger and different sizes were run once (for one seed and one fitness direction criterion only). These include the largest instances available.

Table 7 summarizes these results. As expected, the runtime increases not only with the size of the instance but also for large markets, with higher demand. Nevertheless, considering the strategic scope of the problem at hand, this methodology could still be employed for these extreme cases, probably with a limitation on the number of runs. However, it could be argued that for more complex seasons and markets, such decision-support tools are even more needed. Even if a decision support tool takes 12 hours or a day, it is providing needed support for a season-lasting decision such as fleet size and mix and it may still be useful.

4.5. Variations of the demand function

In order to validate the possibility of using different demand functions in the mathematical programming model proposed, namely in Constraints (5) and (6), and to demonstrate the easiness to adapt the solution methodology, a variation of the demand function was considered. The variation consisted of changing the piece-wise linear shape represented in Fig. 2 to a continuous linear shape, such as the one represented in Fig. 1. This enables the representation of companies where the market position and brand loyalty allow for a "softer" impact of changing pricing positions. Note that even "softer" impacts could be easily modeled using, e.g., an S-curve shape.

The scenario decoder is easily changed, by replacing Eqs. (25) and (26) for the following line equations dependent on price p'. m^* and s^* represent, correspondingly, the slope and intersect values, which are based on previous assumptions plus additional assumptions on the effect on the known price bounds. More specifically, the highest price possible (UBP_r) is assumed to lead to zero demand and the lowest price possible (LBP_r) to the full demand forecast (DF_r).

$$DEM^B_{r\theta}(p') = m^B_{r\theta} \times p' + s^B_{r\theta} \tag{28}$$

$$DEM^A_{r\theta}(p') = m^A_{r\theta} \times p' + s^A_{r\theta} \tag{29}$$

$$\text{where } m^B_{r\theta} = \frac{DF_r \times a_{\theta r}\Delta^{\max}}{LBP_r - COM_{r\theta}}, \quad s^B_{r\theta} = DF_r - m^B_{r\theta} \times LBP_r \tag{30}$$

$$m^A_{r\theta} = \frac{DF_r(1 - a_{\theta r}\Delta^{\max})(1 - b_{\theta r}\Delta^{A-B})}{COM_{r\theta} - UBP_r}, \quad s^A_{r\theta} = -m^A_{r\theta} \times UBP_r \tag{31}$$

The eight instances used for the comprehensive tests were run considering this variation in the demand function (also with ten different seeds and three solution fitness criteria). Table 8 summarizes the main results found.

As far as the final best value is concerned, for the Laplace criteria, for nearly all instances the final best value achieved decrease. With the Pessimist criterion, the trend was opposite depending on whether the market size was small or large, while there were virtually no changes when the Optimist criterion was considered.

The average changes in rental prices allow understanding if these changes in final value result from significant differences in the structure of the solutions obtained. For the Pessimist criterion, the increase/decrease on best value depending on market size can be explained by a corresponding increase/decrease in average prices, which is a direct result of the different demand function

Table 8
Variation of best final value and average rental price.

Instance	Final best value variation vs. original demand function			Average rental price variation vs. original demand function		
	Laplace	Pessimist	Optimist	Laplace	Pessimist	Optimist
1	0%	29%	0%	0%	32%	0%
2	−14%	−15%	0%	−7%	−22%	0%
3	−13%	13%	0%	2%	25%	0%
4	−14%	−19%	0%	−5%	−20%	0%
5	−9%	7%	0%	1%	25%	0%
6	−23%	−24%	0%	−4%	−23%	0%
7	−16%	59%	0%	2%	21%	1%
8	−7%	−7%	0%	−1%	−9%	0%

shape. For the Laplace criterion, however, the percent variations in the rental prices were smaller and did not match in trend the variation in final best value.

This analysis allowed to shed some light on the impact of a different demand function shape for this specific problem, and the impact of the solution fitness criterion selected and the market size considered. Nevertheless, it is important in the future to further analyze this topic in more detail. Also importantly, it demonstrated how to adapt the method to different assumptions easily.

5. Conclusions

This study presents not only a new approach to deal with an innovative application, but also methodological contributions that can be applied beyond this scope. This methodology can be adapted to provide good solutions to complex two-stage stochastic problems where the information on uncertainty is scarce. It does not require the decision-maker to define the scenarios or probabilities associated with them, but only to establish upper and lower bounds for the uncertain parameters. The scenarios are generated and evolve alongside the solutions, and are fine-tuned to be representative and diverse concerning these solutions. This is of particular interest in practical applications where the number of uncertain parameters is large and the explicit definition of uncertainty scenarios is difficult to obtain. Moreover, this method provides the decision-maker with a set of possible solutions, clearly associated with the impact that the different scenarios have on them.

In the future, regarding this innovative application, the model can be extended towards a more tactical (possibly weekly) scope, in order to develop a decision-making support tool for the decisions not considered here. In the car rental problem, this could refer to decisions regarding multi-stage-oriented pricing and deployment actions. Additionally, the work here developed considering the car rental business can be extended to be applied in innovative mobility systems such as car sharing, where the impact of the pricing-demand relationship on the capacity decisions poses some similarities.

Moreover, this methodology can be further developed. If adapted to thoroughly-tested problems with known analytical solutions, intensive computational tests can help improve the efficiency of the genetic algorithm regarding the quality of the solutions achieved and runtime, as well as to validate the conclusions drawn in this work.

Acknowledgments

This work is partially financed by the ERDF European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation – COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT – Fundação para a Ciência e a Tecnologia within project "POCI-01-0145-FEDER-029279".

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ejor.2019.01.015.

References

Adida, E., & Perakis, G. (2010). Dynamic pricing and inventory control: Robust vs. stochastic uncertainty models – A computational study. *Annals of Operations Research, 181*(1), 125–157. doi:10.1007/s10479-010-0706-1.

Ainscough, T. L., Trocchia, P. J., & Gum, J. R. (2009). Consumer rental car choice: Price, agent, and brand effects. *Journal of Business & Economics Research, 7*(7), 39–58.

Bimpikis, K., Candogan, O., & Saban, D. (2016). Spatial pricing in ride-sharing networks. https://ssrn.com/abstract=2868080.

Chemla, D., Meunier, F., Pradeau, T., Calvo, R. W., & Yahiaoui, H. (2013). Self-service bike sharing systems: Simulation, repositioning, pricing. *Technical Report hal-00824078*. Centre d'Enseignement et de Recherche en Mathématiques et Calcul Scientifique CERMICS, Laboratoire d'Informatique de Paris-Nord LIPN, Parallélisme, Réseaux, Systèmes d'information, Modélisation PRISM (2013). Available at: https://hal.archives-ouvertes.fr/hal-00824078.

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2002). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. *IEEE Transactions on Evolutionary Computation, 6*(2), 182–197. doi:10.1007/3-540-45356-3_83.

Den Boer, A. V. (2015). Dynamic pricing and learning: Historical origins, current research, and new directions. *Surveys in Operations Research and Management Science, 20*(1), 1–18. doi:10.1016/j.sorms.2015.03.001.

Fink, A., & Reiners, T. (2006). Modeling and solving the short-term car rental logistics problem. *Transportation Research Part E: Logistics and Transportation Review, 42*(4), 272–292. doi:10.1016/j.tre.2004.10.003.

Freund, D., Henderson, S. G., & Shmoys, D. B. (2017). Minimizing multimodular functions and allocating capacity in bike-sharing systems. In F. Eisenbrand, & J. Koenemann (Eds.), *Proceedings of the integer programming and combinatorial optimization, IPCO 2017: 10328* (pp. 186–198). Springer, Cham. doi:10.1007/978-3-319-59250-3_16.

Gallego, G., & Hu, M. (2014). Dynamic pricing of perishable assets under competition (about demand). *Management Science, 60*(5), 1–45. doi:10.2139/ssrn.1308848.

Gallego, G., & van Ryzin, G. (1994). Optimal dynamic pricing of inventories with stochastic demand over finite horizons. *Management Science, 40*(8), 999–1020. doi:10.1287/mnsc.40.8.999.

Gao, W., Nallaperuma, S., & Neumann, F. (2016). Feature-based diversity optimization for problem instance classification. In Springer (Ed.), *Proceedings of the international conference on parallel problem solving from nature* (pp. 869–879). doi:10.1007/978-3-642-32937-1.

Gonçalves, J. F., & Resende, M. G. C. (2011). Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics, 17*(5), 487–525. doi:10.1007/s10732-010-9143-1.

Gu, J., Gu, M., Cao, C., & Gu, X. (2010). A novel competitive co-evolutionary quantum genetic algorithm for stochastic job shop scheduling problem. *Computers and Operations Research, 37*(5), 927–937. doi:10.1016/j.cor.2009.07.002.

Guastaroba, G., Mansini, R., & Speranza, M. G. (2009). On the effectiveness of scenario generation techniques in single-period portfolio optimization. *European Journal of Operational Research, 192*(2), 500–511. doi:10.1016/j.ejor.2007.09.042.

Herrmann, J. W. (1999). A genetic algorithm for minimax optimization problems. In *Proceedings of the 1999 congress on evolutionary computation, CEC: 2* (pp. 1099–1103). doi:10.1109/CEC.1999.782545.

Høyland, K., & Wallace, S. W. (2001). Generating scenario trees for multistage decision problems. *Management Science, 47*(2), 295–307. doi:10.1287/mnsc.47.2.295.9834.

Jensen, M. T. (2001). A new look at solving minimax problems with coevolution. In *Proceedings of the fourth metaheuristics international conference, MIC'2001* (pp. 103–107).

Kaut, M., & Wallace, S. W. (2003). Evaluation of scenario-generation methods for stochastic programming. *Stochastic Programming E-Print Series, 14*(1).

Krasko, V., & Rebennack, S. (2017). Two-stage stochastic mixed-integer nonlinear programming model for post-wildfire debris flow hazard management: Mitigation and emergency evacuation. *European Journal of Operational Research, 263*(1), 265–282. doi:10.1016/j.ejor.2017.05.004.

Löhndorf, N. (2016). An empirical analysis of scenario generation methods for stochastic optimization. *European Journal of Operational Research, 255*(1), 121–132. doi:10.1016/j.ejor.2016.05.021.

Lu, J., Gupte, A., & Huang, Y. (2018). A mean-risk mixed integer nonlinear program for transportation network protection. *European Journal of Operational Research, 265*(1), 277–289. doi:10.1016/j.ejor.2017.07.025.

Mitra, S., & Di Domenica, N. (2010). A review of scenario generation methods. *International Journal of Computing Science and Mathematics (IJCSM), 3*(3), 226–244. doi:10.1504/IJCSM.2010.037446.

Oliveira, B. B., Carravilla, M. A., & Oliveira, J. F. (2017a). A dynamic programming approach for integrating dynamic pricing and capacity decisions in a rental context. In A. I. F. Vaz, J. Almeida, J. F. Oliveira, & A. A. Pinto (Eds.), *Proceedings of the Portuguese operational research society, APDIO 2017*. In *Springer proceedings in mathematics & statistics: 223* (pp. 297–311). Springer, Cham. doi:10.1007/978-3-319-71583-4.

Oliveira, B. B., Carravilla, M. A., & Oliveira, J. F. (2017b). Capacity-pricing model: Car rental instances. *Mendeley Data, V1*.

Oliveira, B. B., Carravilla, M. A., & Oliveira, J. F. (2017c). Fleet and revenue management in car rental companies: A literature review and an integrated conceptual framework. *Omega, 71*, 11–26. doi:10.1016/j.omega.2016.08.011.

Oliveira, B. B., Carravilla, M. A., & Oliveira, J. F. (2018). Integrating pricing and capacity decisions in car rental: A matheuristic approach. *Operations Research Perspectives, 5*, 334–356. doi:10.1016/j.orp.2018.10.002.

Özcan, U. (2010). Balancing stochastic two-sided assembly lines: A chance-constrained, piecewise-linear, mixed integer program and a simulated annealing algorithm. *European Journal of Operational Research, 205*(1), 81–97. doi:10.1016/j.ejor.2009.11.033.

Pachon, J., Iakovou, E., & Ip, C. (2006). Vehicle fleet planning in the car rental industry. *Journal of Revenue and Pricing Management, 5*(3), 221–236. doi:10.1057/palgrave.rpm.5160041.

Pachon, J., Iakovou, E., Ip, C., & Aboudi, R. (2003). A synthesis of tactical fleet planning models for the car rental industry. *IEE Transactions, 35*(9), 907–916. doi:10.1080/07408170390220674.

Puga, M. S., & Tancrez, J.-S. (2017). A heuristic algorithm for solving large location inventory problems with demand uncertainty. *European Journal of Operational Research, 259*(2), 413–423. doi:10.1016/j.ejor.2016.10.037.

Reis, L. F. R., Walters, G. A., Savic, D., & Chaudhry, F. H. (2005). Multi-reservoir operation planning using hybrid genetic algorithm and linear programming (GA-LP): An alternative stochastic approach. *Water Resources Management, 19*(6), 831–848.

Toso, R. F., & Resende, M. G. C. (2015). A C++ application programming interface for biased random-key genetic algorithms. *Optimization Methods and Software, 30*(1), 1–16. doi:10.1080/10556788.2014.890197.

Wang, K.-J., Makond, B., & Liu, S. (2011). Expert systems with applications location and allocation decisions in a two-echelon supply chain with stochastic demand: A genetic-algorithm based solution. *Expert Systems With Applications, 38*(5), 6125–6131. doi:10.1016/j.eswa.2010.11.008.

Yan, S., Tang, C. H., & Fu, T. C. (2008). An airline scheduling model and solution algorithms under stochastic demands. *European Journal of Operational Research, 190*(1), 22–39. doi:10.1016/j.ejor.2007.05.053.