Transportation Research Part B 81 (2015) 461–482

Contents lists available at ScienceDirect

Transportation Research Part B

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

Trip pricing of one-way station-based carsharing networks with zone and time of day price variations

Diana Jorge [a,*], Goran Molnar [b,1], Gonçalo Homem de Almeida Correia [c,2]

[a] Department of Civil Engineering, University of Coimbra, 3030-788 Coimbra, Portugal
[b] Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, 10000 Zagreb, Croatia
[c] Department of Transport and Planning, Delft University of Technology, P.O. Box 5048, 2600 GA Delft, The Netherlands

ARTICLE INFO

Article history:
Received 15 September 2014
Received in revised form 21 April 2015
Accepted 6 June 2015
Available online 2 July 2015

Keywords:
One-way carsharing
Trip pricing
Mathematical programming
Iterated local search

ABSTRACT

One-way station-based carsharing systems provide short term car rentals in which users can take a car from the initial station and return it to any other station. They are more flexible than round-trip carsharing, where the vehicle can only be returned to the station where it was picked up, and can be used for daily commuting trips as well. This flexibility, however, comes at a cost of vehicle stock imbalance within the network. Several solutions and strategies have been suggested to counter this problem, one of which is variable trip pricing. By charging high prices for the trips that increase imbalance and lowering prices for trips that help improve the balance, it has been hypothesized, but never demonstrated, that the clients' behavior could be used to balance the vehicle stocks and thus make carsharing systems more manageable and profitable. In this paper, we develop a mixed integer non-linear programming (MINLP) model, defined as the Trip Pricing Problem for One-Way Carsharing Systems (TPPOCS), which sets these prices in order to maximize profit. An iterated local search (ILS) metaheuristic is proposed for solving it. The method is applied to the theoretical case-study of a network of 75 stations distributed across the city of Lisbon (Portugal). Although the implemented metaheuristic is tuned for the Lisbon example, the generic nature of its operators makes the model applicable elsewhere. The results demonstrate that the trip pricing strategy can be used to increase profit through a more balanced system. If no price-based balancing strategies are applied, operating this service results in a daily deficit of €1161. When the trip pricing policy is applied, profits of 2068 €/day are possible. The optimal prices are on average 23% higher than the base price, and 18% less demand is served, but the enhanced performance leads to lower expenses with the fleet of vehicles and number of parking spaces.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Carsharing systems involve a small to medium fleet of vehicles available at several stations or parking places spread across a city, which can be used by a relatively large group of members (Shaheen et al., 1999). These systems appeared in 1948, in Europe (Shaheen et al., 1999), as an alternative to private vehicle use and public transport. They give users access

* Corresponding author. Tel.: +351 239 797 131.
E-mail addresses: diana.jorge@student.dec.uc.pt (D. Jorge), goran.molnar2@fer.hr (G. Molnar), G.Correia@tudelft.nl (G.H. de Almeida Correia).
[1] Tel.: +385 1 6129 967.
[2] Tel.: +31 15 27 81384.

http://dx.doi.org/10.1016/j.trb.2015.06.003
0191-2615/© 2015 Elsevier Ltd. All rights reserved.

to a fleet of vehicles, with whatever flexibility and accessibility they provide, without the expense of owning them. Vehicles in these systems also have a higher utilization rate than privately owned ones (Litman, 2000; Schuster et al., 2005; U.S. Department of Transportation, 2001). At the same time, they can help to mitigate negative externalities associated with cars, by reducing the amount of pollutant emissions, for example, especially if an eco-fleet of vehicles is used. Compared to traditional public transport systems, such as bus and metro, carsharing systems can serve to expand service coverage and provide more schedule flexibility.

In the 1980s, successful programs were launched in Europe and carsharing was expanded to the United States, gaining popularity in this country only in the 1990s (Shaheen et al., 1999). Carsharing is now implemented all over the world, which has motivated several studies analyzing its performance and contribution to the accessibility of urban activities (Celsor and Millard-Ball, 2007; Firnkorn and Müller, 2011; Martin and Shaheen, 2011; Schure et al., 2012; Sioui et al., 2013; Lopes et al., 2015).

Regarding the operating model, carsharing systems can be classified as *round-trip* systems, in which users have to return the vehicles to the same station where they were picked up, and *one-way* systems that give users the flexibility to return the vehicle to a different station from the one where it was picked up (Shaheen et al., 2006). Moreover, a particular type of one-way carsharing has appeared recently, the so-called *free-floating* carsharing, whereby vehicles may be picked up and left in parking spaces spread across a city (Ciari et al., 2014; Schmoller et al., 2014).

Round-trip carsharing is generally used for a limited number of purposes, for example shopping, leisure and occasional trips (Barth and Shaheen, 2002; Costain et al., 2012), while one-way carsharing is used for all purposes, including commuting (Balac and Ciari, 2014; Ciari et al., 2014). However, there is a major drawback to one-way carsharing: a station's vehicle stocks often become imbalanced, constituting a problem for the service operator (Kek et al., 2009; Febbraro et al., 2012; Jorge and Correia, 2013). This happens because demand for vehicles varies during the day and for origin–destination station pairs, creating a tendency for vehicles to accumulate at stations where they are not needed and therefore causing a shortage in stations where there is demand. This situation can even lead to a parking problem for users wanting to park their vehicle at a saturated station. The management of parking space reservation policies to cope with this problem has in fact been addressed in a study by Kaspi et al. (2014).

Several approaches have been proposed in the literature to mitigate the effects of vehicle stock imbalance where the most extensively studied method is vehicle relocation (Barth and Todd, 1999; Barth et al., 2001; Kek et al., 2006; Kek et al., 2009; Nair and Miller-Hooks, 2011; Krumke et al., 2013; Jorge et al., 2014; Nourinejad and Roorda, 2014; Repoux et al., 2014; Boyaci et al., 2015). Alternative methods were: accepting or refusing a trip based on its impact on vehicle stock balance (Fan et al., 2008; Correia and Antunes, 2012); station location selection to achieve a more favorable distribution of vehicles (Correia and Antunes, 2012), and price incentives for grouping people if they are traveling from a station with a shortage of vehicles, and ungrouping them otherwise (Barth et al., 2004; Uesugi et al., 2007). Regarding the use of pricing we identified two methods in the literature: price incentive policies for users to choose another drop-off station, where total demand stays unaltered (Febbraro et al., 2012; Weikl and Bogenberger, 2013; Pfrommer et al., 2014), and trip pricing, that is, changing the trip prices to control the demand, taking its contribution to stock balancing into account, which is seen as a proxy for profit maximization (Mitchell et al., 2010; Zhou, 2012).

Mitchell et al. (2010) and Weikl and Bogenberger (2013) only suggested this as a theoretical balancing strategy and did not define ways of choosing prices and applying them in reality. In the case of the other authors, the methodological approaches mainly fall into two camps, one in which the state of stations is analyzed to determine if they are oversupplied, undersupplied or balanced, and the other where the method is implemented and the users' response to it is studied, by means of a simulation approach (Febbraro et al., 2012; Zhou, 2012; Pfrommer et al., 2014). Moreover, Febbraro et al. (2012) and Zhou (2012) grouped stations with similar behavior into zones charging prices per pair of zones. All these studies assumed that if users do not accept the proposed destination station or new OD pair of zones they can still make their desired trip for the initial price.

In this paper we want to bridge the apparent research gap in the literature by proposing a method for optimally setting the trip prices in one-way carsharing systems and show how it can be useful for profit maximization by reducing vehicle fleet imbalance. Contrary to previous studies, pricing is not used as a reactive measure but as a stable reference of the company that practices a table of prices that should be tailored to the existing consumer preferences and the operational constraints of the company.

It is relevant to say that even though our methodological approach has not yet been implemented to solve carsharing imbalance problems, it has been used to solve other transportation problems, in particular with respect to: traffic congestion (Wie and Tobin, 1998; Joksimovic et al., 2006; Wang et al., 2011; Chung et al., 2012); high occupancy/toll lane management (Lou et al., 2011), and airline seat management (Schon, 2008; Atasoy et al., 2014). As in our approach, these methods also consider elastic travel demand towards price either with simple elasticity models or expressed by logit models (Schon, 2008; Lou et al., 2011; Atasoy et al., 2014). Logit models are more precise but they also make it more difficult to reach an optimized solution, since they introduce a non-linear and non-convex behavior of the objective function. To address this limitation, researchers used three main methods: transforming the non-convex formulation into a convex one (Wie and Tobin, 1998; Schon, 2008), for example by using the inverse of the logit model function (Schon, 2008); using heuristics (Joksimovic et al., 2006) or metaheuristics (Chung et al., 2012; Atasoy et al., 2014); or using simulation instead of optimization (Lou et al., 2011).

The method that we developed and present in this paper has two main components: (*i*) a mixed-integer non-linear programming (MINLP) model which, with trips made throughout an entire day and the price elasticity of demand, determines which prices to charge in a given period of time for profit maximization; (*ii*) an *iterated local search* (ILS) metaheuristic to find good solutions as fast as possible, given the non-linearity of the MINLP.

We decided to maximize profit since carsharing is mostly provided by private companies. Nevertheless, it should be noted that maximizing profit, although not leading to the highest level of service provided to the clients, it should allow to operate a bigger network since the company is able to profit even when its market is highly imbalanced as it happens often with encompassing city center and suburban areas (Correia and Antunes, 2012; Jorge et al., 2015).

The method is applied and tested for the case study of Lisbon, in Portugal, providing the following main scientific contributions:

- it is the first optimized method to set a variable trip pricing table for one-way carsharing;
- it is a method that leads to better results than having no balancing strategy and it avoids high logistic adaptations entailed in using relocation operations.

The paper is structured as follows. In the next section, we present the method used to address this problem. In Section 3, the metaheuristic solution algorithm that is part of the method is explained. Section 4 describes the case study to which the method is applied. Next, the computational experiments on the case-study are defined in Section 5, and the main results are presented in Section 6. Finally, the paper set out the most important conclusions drawn from this study and some possibilities for future development.

For reference, the acronyms used throughout the paper are summarized in Table 1.

2. Method

In this section, as a basis for the method of setting the trip prices, we first introduce the model proposed and applied by Jorge et al. (2014) in which optimal relocation movements are found for a set of stations in a one-way carsharing system. This model is able to provide a desirable theoretical vector of relocated vehicles between each pair of stations, once the staff relocation costs are set to 0, which is further used as a reference for the new trip pricing model.

The new model, which uses prices for managing vehicle stocks, is adapted from the previous one, incorporating price variations per pair of zones and period of the day, replacing the reference static price with a new set of decision variables and introducing price elasticity of demand. Relocation operations are naturally excluded from the new model. A cluster analysis is run to find groups of stations using the desirable theoretical vector of relocated vehicles as clustering information. This vector was derived from the model by Jorge et al. (2014), with zero relocation costs as noted above. Given the dependence of demand with respect to price, the new model becomes non-linear, which makes it difficult to solve. At the end of this section we propose a lower and upper bounds computation for the optimum value of the objective function. This will provide a reference for then solving the problem using a metaheuristic method, which is explained in the next section.

The method assumes that the estimated value of the price elasticity of demand is known and that the mobility patterns in the network can be forecast. Elasticity is typically estimated from historical sales information and can be extracted using different statistical methods (Bordley, 1994; Taplin et al., 1999; Fouquet, 2012). Transportation forecasting can also be done by simulation or modeling based on the data acquired from past network use or from surveying users. Since there is ample research on this separate subject (de Dios Ortúzar and Willumsen, 2011; Sinha and Labi, 2007), a detailed description of the techniques for demand modeling is beyond the scope of this paper.

2.1. The Vehicle Relocation Problem for One-way Carsharing Systems (VRPOCS)

Jorge et al. (2014) proposed an integer linear programming (ILP) model for the optimal relocation movements between a set of stations to maximize the daily profit of a one-way carsharing company. We denote this as the Vehicle Relocation

Table 1
Acronyms used in the paper.

Abbreviation	Complete form
CBD	Central business district
ILP	Integer linear programming
ILS	Iterated local search
LSO	Local search operator
MINLP	Mixed-integer non-linear programming
MIP	Mixed integer programming
OD	Origin–destination
PO	Perturbation operator
TI	Time interval
TPPOCS	Trip pricing problem for one-way carsharing systems
VRPOCS	Vehicle relocation problem for one-way carsharing systems

Problem for One-way Carsharing Systems (VRPOCS). These movements are assumed to be performed by a staff of drivers and all demand between stations has to be satisfied.

The notation used to formulate the model (sets, data and decision variables) is as follows:

Sets:

$\mathbf{K}' = \{1, \ldots, k \ldots K\}$: Set of stations.

$\mathbf{T}' = \{1, \ldots, t \ldots T\}$: Set of time instants in the operation period.

$\mathbf{X} = \{1_1, \ldots, k_{t-1}, k_t, k_{t+1}, \ldots, K_T\}$: Set of nodes of a time–space network combining the K stations with the T time instants, where k_t represents station k at time instant t.

$\mathbf{A}_1 = \left\{\ldots, \left(k_t, j_{t+\delta^t_{kj}}\right), \ldots\right\}, k_t \in \mathbf{X}$: Set of arcs over which vehicles move between stations k and j, $\forall k, j \in \boldsymbol{K}', k \neq j$, between time instant t and $t + \delta^t_{kj}$.

$\mathbf{A}_2 = \{\ldots, (k_t, k_{t+1}), \ldots\}, k_t \in \mathbf{X}$: Set of arcs that represent vehicles stocked in station k, $\forall k \in \boldsymbol{K}'$, from time instant t to time instant $t+1$.

Data:

δ^t_{kj}: Travel time, in time instants, between stations k and j when departure time is t, $\forall k_t \in \boldsymbol{X}, j \in \boldsymbol{K}'$.

$P0$: The current carsharing price for all OD pairs of stations at any time instant.

C_{mv}: The maintenance cost of each vehicle per time step driven.

C_{mp}: The cost of maintaining one parking space per day.

C_v: The depreciation cost of one vehicle per day.

C_r: The relocation cost per vehicle per time step driven.

$D0_{k_t j_{t+\delta^t_{kj}}}$: Number of customer trips from station k to station j from instant t to instant $t + \delta^t_{kj}$, $\forall \left(k_t, j_{t+\delta^t_{kj}}\right) \in \boldsymbol{A}_1$ for the reference price.

Decision variables:

$R_{k_t j_{t+\delta^t_{kj}}}$: Number of vehicles relocated from k to j from time instant t to $t + \delta^t_{kj}$, $\forall \left(k_t, j_{t+\delta^t_{kj}}\right) \in \boldsymbol{A}_1$.

Z_k: Size of station k, $\forall k \in \boldsymbol{K}'$, where size refers to the number of parking spaces.

a_{k_t}: Number of available vehicles at station k at time instant t, $\forall k_t \in \boldsymbol{X}$.

Auxiliary variable:

$V_{k_t k_{t+1}}$: Number of vehicles stocked at each station k from time instant t to $t+1$, $\forall (k_t, k_{t+1}) \in \boldsymbol{A}_2$.

Using the above notation, the model is formulated as follows:

$$Max\ \pi = (P0 - C_{mv}) \times \sum_{k_t j_{t+\delta^t_{kj}} \in \boldsymbol{A}_1} D0_{k_t j_{t+\delta^t_{kj}}} \times \delta^t_{kj} - C_{mp} \sum_{k \in \boldsymbol{K}'} Z_k - C_v \sum_{k \in \boldsymbol{K}'} a_{k_1} - C_r \sum_{k_t j_{t+\delta^t_{kj}} \in \boldsymbol{A}_1} R_{k_t j_{t+\delta^t_{kj}}} \times \delta^t_{kj} \tag{1}$$

subject to,

$$V_{k_t k_{t+1}} + \sum_{j_t \in \mathbf{X}} D0_{k_t j_{t+\delta^t_{kj}}} + \sum_{j_t \in \mathbf{X}} R_{k_t j_{t+\delta^t_{kj}}} - \sum_{j \in \mathbf{K}' : t' = t - \delta^t_{jk}} D0_{j_{t'} k_t} - \sum_{j \in \mathbf{K}' : t' = t - \delta^t_{jk}} R_{j_{t'} k_t} - V_{k_{t-1} k_t} = 0 \ \forall k_t \in \mathbf{X} \tag{2}$$

$$a_{k_t} - \sum_{j_t \in \mathbf{X}} D0_{k_t j_{t+\delta^t_{kj}}} - \sum_{j_t \in \mathbf{X}} R_{k_t j_{t+\delta^t_{kj}}} - V_{k_t k_{t+1}} = 0 \ \forall k_t \in \mathbf{X} \tag{3}$$

$$Z_k \geqslant a_{k_t}, \ \forall k_t \in \mathbf{X} \tag{4}$$

$$R_{k_t j_{t+\delta^t_{kj}}} \in \mathbb{N}^0, \ \forall \left(k_t, j_{t+\delta^t_{kj}}\right) \in \mathbf{A}_1 \tag{5}$$

$$V_{k_t k_{t+1}} \in \mathbb{N}^0, \ \forall (k_t, k_{t+1}) \in \mathbf{A}_2 \tag{6}$$

$$a_{k_t} \in \mathbb{N}^0, \ \forall k_t \in \mathbf{X} \tag{7}$$

$$Z_k \in \mathbb{N}^0, \ \forall k \in \mathbf{K}' \tag{8}$$

The objective function (1) is to maximize the total daily profit (π) of the one-way carsharing service, taking into consideration the revenue from the trips paid by clients, vehicle maintenance costs, vehicle depreciation costs, station maintenance costs, and relocation costs. Constraints (2) ensure the conservation of vehicle flows at each node of the time–space network. Constraints (3) compute the number of vehicles at each station k at the start of time instant t, assuming that vehicles destined to arrive at station k at time instant t arrive before vehicles leave from the same station at time instant t. Constraints (4) guarantee that the size of the station at location k is greater than the number of vehicles located there at each time instant t. In practice, size will not be greater than the largest value of a_{k_t} during the period of operation because this would penalize the objective function. Expressions (5)–(8) establish that the variables must be integer and positive.

2.2. Grouping stations through k-means clustering

Running the VRPOCS model for null relocation operations costs ($C_r = 0$) yields an ideal vector of relocation flows. With this vector it is possible to compute the ideal number of relocated vehicle entries and exits at each station.

The notation introduced in this sub-section is as follows:

Sets:

$\mathbf{I}' = \{1, \ldots, i \ldots I\}$: The set of time intervals in the operation period.

$\mathbf{Z}' = \{1, \ldots, z \ldots Z\}$: The set of zones.

Parameters:

$\alpha^0_{k_t}$: Number of vehicles relocated to station k at time instant t when relocation costs are 0, $\forall k_t \in \boldsymbol{X}$.

$\beta^0_{k_t}$: Number of vehicles relocated from station k at time instant t when relocation costs are 0, $\forall k_t \in \boldsymbol{X}$.

$\epsilon^0_{k_t}$: Difference in the number of vehicles relocated to/from station k at time instant t when relocation costs are 0, $\forall k_t \in \boldsymbol{X}$.

tb_i: The beginning instant of time interval i, $\forall i \in \boldsymbol{I}'$.

te_i: The end instant of time interval i, $\forall i \in \boldsymbol{I}'$.

$\omega^0_{k_i}$: Difference in the number of vehicles relocated to/from station k during time interval i when relocation costs are 0, $\forall k \in \boldsymbol{K}'$, $\forall i \in \boldsymbol{I}'$.

o: Number of observations in the cluster analysis.

u: Number of clusters desired.

Using the notation presented above:

– the vehicle entries are given by:

$$\alpha^0_{k_t} = \sum_{j \in \mathbf{K}' : t' = t - \delta^t_{jk}} R_{j_{t'} k_t}, \ \forall (t', k_t) \in \mathbf{A}_1 \tag{9}$$

– the vehicle exits are given by:

$$\beta^0_{k_t} = \sum_{j_t \in \boldsymbol{X}} R_{k_t j_{t+\delta^t_{kj}}}, \ \forall \left(k_t, j_{t+\delta^t_{kj}}\right) \in \boldsymbol{A}_1 \tag{10}$$

The difference between the vectors $\alpha^0_{k_t} - \beta^0_{k_t}$ yields a new vector $\epsilon^0_{k_t}$ that will be positive or negative depending on whether station i is a supplier ($\epsilon^0_{k_t} > 0$) or a demander of vehicles ($\epsilon^0_{k_t} < 0$).

The values of the vector $\epsilon^0_{k_t}$ can be aggregated for each time interval. We call the set of time intervals into which the day is divided the $\mathbf{I}' = \{1, \ldots, i \ldots I\}$ set. Given this, it is possible to compute the vector for the relocation balance at each time interval i as:

$$\omega^0_{k_i} = \sum_{t = tb_i}^{te_i} \epsilon^0_{k_t}, \ \forall k \in \boldsymbol{K}', \quad \forall i \in \boldsymbol{I}' \tag{11}$$

The zones of stations that are similar in their vehicle needs or in their ability to provide vehicles at a given interval are obtained by applying the *K-means* clustering algorithm. *K-means* partitions *o* observations into *u* clusters. Assuming *o* observations, it first chooses *u* centroids, where *u* is the number of clusters desired. Each observation *o* will then be assigned to the closest centroid, and each group of observations assigned to a centroid will be a cluster. The centroid of each cluster is then updated based on the observations assigned to the cluster. We use vector $\omega^0_{k_i}$ as a measure of similarity between the stations in each time interval *i*. Thus the number of *u* centroids is the number of desired zones (set $\boldsymbol{Z}'$) for which prices will vary, and the number of observations *o* is the number of stations (set $\boldsymbol{K}'$). It is well known that this clustering process does not lead to a global optimum, since the process is dependent on the choice of the *u* first observations (Ji and Geroliminis, 2012); nevertheless, this is not the major concern of the method developed herein.

The clustering process described is meant to produce the same number of zones for any of the time intervals and does not require continuity, that is, stations in each zone do not need to be contiguous. A station may belong to one cluster and afterwards to another, which enables any station to have different prices from its neighbor during the day.

2.3. The trip pricing problem for one-way carsharing systems (TPPOCS)

The mixed-integer non-linear programming (MINLP) model proposed in this section derives from the VRPOCS. This problem is defined as follows: *given a set of carsharing stations operating in one-way mode for which an OD matrix is known for a given reference price, the TPPOCS aims at finding new prices between groups of stations during a working day such that the profit of running the system is maximized while satisfying all demand.*

In addition to the notation previously defined, we need the following to implement the MINLP:

Data:
E: Price elasticity of demand.
$P0^i_{zw}$: The current carsharing price per time step driven between zones z and w when departure time interval is i, $\forall z, w \in \boldsymbol{Z}', i \in \boldsymbol{I}'$ (all prices set to $P0$).
Decision variables:
$D_{k_t j_{t+\delta^t_{kj}}}$: Number of customer trips from station k to station j from time instant t to $t+\delta^t_{kj}$, $\forall \left(k_t, j_{t+\delta^t_{kj}}\right) \in \boldsymbol{A}_1$ after the price is varied.
P^i_{zw}: Carsharing price per time step driven between zones z and w when departure time period is i, $\forall z, w \in \boldsymbol{Z}', i \in \boldsymbol{I}'$.

Demand, in this model, varies according to a simple elastic behavior. The new demand $\left(D_{k_t j_{t+\delta^t_{kj}}}\right)$ results from applying the price elasticity E to a reference demand $\left(D0_{k_t j_{t+\delta^t_{kj}}}\right)$ that exists for price $P0$. The expression is:

$$E = \frac{\frac{D_{k_t j_{t+\delta^t_{kj}}} - D0_{k_t j_{t+\delta^t_{kj}}}}{D0_{k_t j_{t+\delta^t_{kj}}}}}{\frac{P^i_{zw} - P0^i_{zw}}{P0^i_{zw}}} \tag{12}$$

We are assuming the elasticity to be the same for any interval of price variation, but this may be unrealistic for large price variations. However, one does not expect to change prices beyond a realistic interval around the current reference price P0.

Using the notation presented in the previous sub-sections and the elasticity defined in Eq. (12), the MINLP model is formulated as follows:

$$Max\ \theta = \sum_{\substack{k_t j_{t+\delta^t_{kj}} \in \boldsymbol{A}_1 \\ z,w \in \boldsymbol{Z}' \\ i \in \boldsymbol{I}'}} \left(P^i_{zw} - C_{mv}\right) \times D_{k_t j_{t+\delta^t_{kj}}} \times \delta^t_{kj} - C_{mp} \sum_{k \in \boldsymbol{K}'} Z_k - C_v \sum_{k \in \boldsymbol{K}'} a_{k_1} \tag{13}$$

subject to,

$$D_{k_t j_{t+\delta^t_{kj}}} \geq D0_{k_t j_{t+\delta^t_{kj}}} + \frac{E \times D0_{k_t j_{t+\delta^t_{kj}}} \times \left(P^i_{zw} - P0^i_{zw}\right)}{P0^i_{zw}} - 0.5,\ \forall \left(k_t, j_{t+\delta^t_{kj}}\right) \epsilon A_1,\ z, w \in \mathbf{Z}',\ i \in \mathbf{I}' \tag{14}$$

$$D_{k_t j_{t+\delta^t_{kj}}} \leq D0_{k_t j_{t+\delta^t_{kj}}} + \frac{E \times D0_{k_t j_{t+\delta^t_{kj}}} \times \left(P^i_{zw} - P0^i_{zw}\right)}{P0^i_{zw}} + 0.5,\ \forall \left(k_t, j_{t+\delta^t_{kj}}\right) \epsilon A_1,\ z, w \in \mathbf{Z}',\ i \in \mathbf{I}' \tag{15}$$

$$D0_{k_t j_{t+\delta^t_{kj}}} + \frac{E \times D0_{k_t j_{t+\delta^t_{kj}}} \times \left(P^i_{zw} - P0^i_{zw}\right)}{P0^i_{zw}} \geq 0 \tag{16}$$

$$V_{k_t k_{t+1}} + \sum_{j \in \boldsymbol{K}'} D_{k_t j_{t+\delta^t_{kj}}} - \sum_{j \in \boldsymbol{K}' : t' = t - \delta^t_{jk}} D_{j_{t'} k_t} - V_{k_{t-1} k_t} = 0, \quad \forall k_t \in \mathbf{X} \tag{17}$$

$$a_{k_t} - \sum_{j_t \in \mathbf{X}} D_{k_t j_{t+\delta^t_{kj}}} - V_{k_t k_{t+1}} = 0, \quad \forall k_t \in \mathbf{X} \tag{18}$$

$$Z_k \geq a_{k_t}, \quad \forall k_t \in \mathbf{X} \tag{19}$$

$$D_{k_t j_{t+\delta^t_{kj}}} \in \mathbb{N}^0,\ \forall \left(k_t, j_{t+\delta^t_{kj}}\right) \in \mathbf{A}_1 \tag{20}$$

$$P^i_{zw} \in \mathbb{R}^0,\ \forall z, w \in \mathbf{Z}',\ i \in \mathbf{I}' \tag{21}$$

$$V_{k_t k_{t+1}} \in \mathbb{N}^0, \quad \forall (k_t, k_{t+1}) \in \mathbf{A}_2 \tag{22}$$

$$a_{k_t} \in \mathbb{N}^0,\ \forall k_t \in \mathbf{X} \tag{23}$$

$$Z_k \in \mathbb{N}^0,\ \forall k \in \mathbf{K}' \tag{24}$$

The objective function (13) maximizes the total daily profit (θ) of the one-way carsharing service, taking into consideration the revenue from trips paid by the clients, vehicle maintenance costs, vehicle depreciation costs, and station maintenance costs. Note that in this model no relocations are considered. Constraints (14) and (15) compute the demand resulting from considering the price change. Given that this demand is a continuous function of price, we use two inequalities to ensure that D will be integer. Constraints (16) ensure that the demand resulting from the application of price elasticity to the reference demand is positive. Constraints (17) and (18) are the same as constraints (2) and (3) from the VRPOCS model, but excluding the variables related to the vehicle relocation operations. Constraints (19) are the same as constraints (4). Expressions (20)–(24) set the variables domain.

The decision variables of the model are: the number of vehicles in each station at the beginning of the day, the demand for each OD pair of stations at each time step, and the prices charged for each OD pair of zones per time interval. We can see that the objective function (13) is non-linear because demand is multiplied by the price and is non-concave, which makes this a MINLP problem not easily solvable by traditional branch and cut algorithms. Some MINLP solver software solutions are available to solve this type of problem for both concave and non-concave formulations, but these typically have difficulties managing real size problem instances (Bussieck and Vigerske, 2014). The size of the search space of our problem is much greater than these solvers are able to tackle. For only 5 zones and 6 time periods, if prices vary from 0 to 0.70 €/min, with 0.01 increments, the number of possible solutions for this problem would be $|\mathcal{P}| = 71^{5\cdot5\cdot6} = 4.88 \cdot 10^{277}$.

2.4. Bound analysis for the TPPOCS problem

As noted, the main limitation of the TPPOCS is that the objective function is quadratic and non-concave, which is why we employ a metaheuristic to solve the problem. We have no guarantee that this solution will be optimal therefore we provide a bound analysis of the objective function for the TPPOCS.

First, we define the calculation of the theoretical lowest value the profit can take. Then, as system imbalance is identified as the major reason for high operating costs, the lower bound of the optimum is defined as the profit of a solution that is as balanced as possible. The upper bound of the profit is computed as the maximum theoretical value of the optimum and we provide a way to calculate it using model simplification.

2.4.1. Lowest profit

Profit, as the difference between the revenues and the operating costs, is the lowest possible when the revenue is lowest and the costs are highest. Lowest revenue clearly occurs when the price of traveling between the origin and destination pairs is zero. In our model, the operating costs rise as the demand increases. This is because greater demand calls for more vehicles and more parking space to accommodate them. The demand is higher when the prices are lower and vice versa. Therefore the free-rides solution satisfies both criteria for the minimum theoretical profit of our problem: they minimize revenue and maximize cost. Simulating the company for the zero trip price therefore gives an easy-to-calculate lowest possible value of our objective function.

2.4.2. Lower bound of the optimum

We propose a new linear objective function for computing the lower bound of the optimal solution profit which avoids the limitations introduced by the previous non-linear, non-concave one.

While it is intuitively clear what imbalance is, formally, it can be defined in different ways. To evaluate imbalance, one could look at utilization rates of parking spaces in the system. If some stations only have 10% of parking spaces occupied most of the time and others are used 100% most of the time, it is a clear indicator that the system is not properly balanced or that something is wrong with the network planning. Likewise, imbalance could be estimated as a difference in the number of incoming and outgoing vehicles at the stations. But even with a strict definition of system balancing for a particular time period, the balance indicator will greatly depend on the period dimension. In this work, we use the latter definition of imbalance by observing differences between vehicle entries and exits for a particular time period.

In the new linear objective function, among other components, we include that balance of vehicle entries and exits for each group of stations (zone) for each period in which the day is divided. The underlying idea is that a balanced system should lead to higher profits because vehicles have a higher probability of being used after arriving to a station.

For solving this lower bound formulation, we need to resort to a new auxiliary variable: y_z^i, which represents the difference between vehicle entries and exits at each zone z at time interval i, $\forall z \in \boldsymbol{Z}', i \in \boldsymbol{I}'$. This variable can take either positive or negative values, because it is defined as a continuous variable in its domain. Since positive values should not be annulled by negative ones, and the squared value cannot be used, two new positive variables are defined: ll_z^i and ul_z^i. ll_z^i and ul_z^i are, respectively, the lower and the upper limit of the y_z^i variation. Taking this into consideration, the objective function should now be redefined and the following constraints should be added to previous model (13)–(24):

$$Min\ \omega = \sum_{\substack{z \in \boldsymbol{Z}' \\ i \in \boldsymbol{I}'}} \left(ll_z^i + ul_z^i \right) - \sum_{\substack{k_t j_{t+\delta_{kj}^t} \in \boldsymbol{A}_1 \\ z,w \in \boldsymbol{Z}' \\ i \in \boldsymbol{I}'}} D_{k_t j_{t+\delta_{kj}^t}} \times \delta_{kj}^t + C_{mp} \sum_{k \in \boldsymbol{K}'} Z_k + C_v \sum_{k \in \boldsymbol{K}'} a_{k_1} \tag{25}$$

Where:

$$-ll_z^i \leqslant y_z^i \leqslant up_z^i \tag{26}$$

$$ll_z^i \geqslant 0 \tag{27}$$

$$ul_z^i \geqslant 0 \tag{28}$$

Objective function (25) adds the balance variables to a changed version of the previous function θ, where prices are not included in the revenues and all the signs are inverted since the objective of this problem is to minimize vehicle imbalance. The point is that this new, simpler to solve, linear model searches for solutions that are at the same time balanced and cheap, and that capture longer trips. There could be many alternatives for the objective function specification. However, with this lower bound formulation we aim to provide a solution that is better than the no-carsharing system trivial solution (no trips, no system), that provides a profit to the *TPPOCS* without the need for a metaheuristic, and that can simultaneously show that the metaheuristic performs significantly better than this simpler but incomplete model. Constraints (26)–(28) set the domain of the new decision variables.

2.4.3. An upper bound of the optimum

For the upper bound we compute the maximum theoretical value of the system profit. Thus, we consider the revenue term of the objective function as well as the vehicle maintenance costs (directly related to the driving distance) and ignore the other cost terms, i.e. number of parking spaces and fleet depreciation, since these form mutual dependences on the constraints of the problem and cannot be simply substituted in the objective function. The search for the set of prices that leads to the maximum value of that objective function (*called* ρ) is considered to be the upper bound. This can then be compared to the results of applying the solution algorithm presented in Section 6.

The revenue function is defined as follows:

$$\rho = \sum_{\substack{k_t j_{t+\delta_{kj}^t} \in \boldsymbol{A}_1 \\ z,w \in \boldsymbol{Z}' \\ i \in \boldsymbol{I}'}} \left(P_{zw}^i - C_{mv}\right) \times D_{k_t j_{t+\delta_{kj}^t}} \times \delta_{kj}^t \tag{29}$$

Substituting $D_{k_t j_{t+\delta_{kj}^t}}$ by $D0_{k_t j_{t+\delta_{kj}^t}} + \frac{2\delta_{kj}^t \times P_{zw}^i \times E \times D0_{k_t j_{t+\delta_{kj}^t}}}{P0_{zw}^i}$ (elasticity expression) in the objective function ρ and deriving it with respect to the vector of prices P_{zw}^i for each I, z, and w, we have the first order condition to find a maximum of the ρ function:

$$\frac{\partial \rho}{\partial P_{zw}^i} = 0,\ \forall z, w \in \boldsymbol{Z}',\ i \in \boldsymbol{I}' \tag{30}$$

$$\sum_{k_t j_{t+\delta_{kj}^t} \in \boldsymbol{A}_1} \left(\delta_{kj}^t \times D0_{k_t j_{t+\delta_{kj}^t}} + \frac{2\delta_{kj}^t \times P_{zw}^i \times E \times D0_{k_t j_{t+\delta_{kj}^t}}}{P0_{zw}^i} - \delta_{kj}^t \times E \times D0_{k_t j_{t+\delta_{kj}^t}} - \frac{\delta_{kj}^t \times C_{mv} \times E \times D0_{k_t j_{t+\delta_{kj}^t}}}{P0_{zw}^i} \right) = 0,\ \forall z, w \in \boldsymbol{Z}',\ i \in \boldsymbol{I}' \tag{31}$$

The summation can disappear and all prices are the same for each OD pair:

$$P_{zw}^i = \frac{-P0_{zw}^i}{2E} + \frac{P0_{zw}^i}{2} + \frac{C_{mv}}{2},\ \forall z, w \in \boldsymbol{Z}',\ i \in \boldsymbol{I}' \tag{32}$$

In order for that maximum to be unique, the second order condition must hold:

$$\frac{\partial^2 \rho}{\partial^2 P_{zw}^i} < 0,\ \forall z, w \in \boldsymbol{Z}',\ i \in \boldsymbol{I}' \tag{33}$$

$$\sum_{k_t j_{t+\delta_{kj}^t} \in \boldsymbol{A}_1} \frac{2\delta_{kj}^t \times E \times D0_{k_t j_{t+\delta_{kj}^t}}}{P0_{zw}^i} < 0,\ \forall z, w \in \boldsymbol{Z}',\ i \in \boldsymbol{I}' \tag{34}$$

Eq. (34) is bounded by the problem constraints to have only negative values because prices have to be positive; demand is constrained to be positive and the elasticity is constant and negative. The set of prices that satisfy the first order conditions will be a global maximum of the ρ function and the maximum theoretical value for the profit is found.

As the price that yields the best revenue is the same for each OD pair, the upper bound calculated this way has the highly important property of being clustering independent. This method will find the same upper estimate for any type of clustering used, even when working directly with prices among stations and no clustering is applied.

3. Solution algorithm

3.1. Iterated local search (ILS)

The goal of the solution algorithm presented in this section is to find the prices P^i_{zw} for which the daily profit θ of the TPPOCS will be as high as possible. A solution of this problem is a set of trip pricing tables denoted $P[|I|][|Z|][|Z|](p_{min}, p_{max})$, or in short $P(p_{min}, p_{max})$, where $|I|$ is the number of time intervals, $|Z|$ is the number of zones and p_{min} and p_{max} are the minimum and maximum allowed prices, respectively. Pricing table $P(p_{min}, p_{max})$ contains $(|I| \cdot |Z| \cdot |Z|)$ individual elements and each element P^i_{zw} corresponds to the price charged per minute for trips from any station in zone z to any station in zone w, starting in time interval$i$. The set of feasible solutions $\mathcal{P}(p_{min}, p_{max})$ is defined as the set of all possible trip pricing tables of appropriate dimensions $(|I| \times |Z| \times |Z|)$ whose elements are in a given price interval.

The optimal pricing table $\bar{P}$ is a trip pricing table for which the daily profit (θ) of a carsharing company is optimal. More formally, the goal of the optimization algorithm is to find $\bar{P}$ for which the following equation is satisfied:

$$Max\ \theta(\overline{P}) \geq Max\ \theta(P), \quad \forall P\ (p_{min},\ p_{max})\ \in \mathcal{P}(p_{min},\ p_{max}). \tag{35}$$

For each trip pricing table $P(p_{min}, p_{max})$ generated by the solution algorithm, the TPPOCS mathematical model is executed as a classical mixed integer programming (MIP) problem where prices are given. In that way, the TPPOCS model finds the best possible profit that can be achieved using a fixed trip pricing table suggested by the solution algorithm. The best possible profit value is then introduced back to the algorithm, in essence rendering the model an evaluator for the solutions suggested by the algorithm.

Various metaheuristic techniques have been very successful at solving this type of problems. They are general problem-independent algorithmic frameworks that can be applied to various optimization contexts. While not guaranteeing optimality, they can, if implemented properly, provide solutions that are certainly good enough for practical use. For many problems, they yield state-of-the-art results (Lourenço et al., 2003; Luke, 2013).

For our problem we used the *iterated local search* (ILS) (Stützle and Hoos, 1999; Lourenço et al., 2001; Lourenço et al., 2003; Luke, 2013) to solve the mixed-integer non-linear TPPOCS. It is a simple but effective metaheuristic that successively employs *local search* (LSO) and *perturbation* operators (PO) in an attempt to focus on exploring proximities of known good solutions while avoiding being stuck in local optima. It has been successfully implemented for classical combinatorial optimization problems, such as traveling salesman problem (Stützle and Hoos, 1999; Katayama and Narihisa, 1999) and the quadratic assignment problem (Stützle, 2006), as well as more specific problems such as scheduling and graph partitioning (Lourenço et al., 2003; Carlier, 1982; Martin and Otto, 1995).

The implementation details of the ILS metaheuristic for solving the TPPOCS are given in the next sections. First, the general structure of the metaheuristic is presented. Second, the algorithm to generate initial solutions is explained. Finally, the implementation details of the LSO and PO are given.

3.2. Algorithm structure

The ILS algorithm is based on two operators (Lourenço et al., 2001; Lourenço et al., 2003; Luke, 2013):

1. *local search* operator (LSO), and
2. *perturbation* operator (PO).

The pseudo-code of the algorithm we use is given in Algorithm 1.

The local search looks for the best solution in a restricted neighborhood of the initial solution. Given an initial solution $P_{initial}$, it enumerates all the solutions from its neighborhood and returns the best as the result, which is called *local optimum* and denoted P^*. The quality of the local optima depends on the way the neighborhood structure is defined and on the initial solution choice.

Although the local optima are as good as or better than the initial solution, in general there is no warranty as to their quality in the context of all possible solutions. Local search results are often globally suboptimal (Lourenço et al., 2003; Luke, 2013), and to extend the search beyond the initial neighborhood, ILS uses the PO. The PO takes the current local optimum and modifies it into a perturbed solution, denoted P'. The intensity of the modification should be low enough to prevent the algorithm from losing focus and degrading to randomly restarting the local search, but high enough to ensure the local search does not converge to the same solution during the next iteration (Lourenço et al., 2003). The local optimum in the environment of the perturbed pricing table P' during the ILS algorithm run is denoted as P'^*.

Algorithm 1. Pseudo-code of the implemented iterated local search (ITS) metaheuristic.

Procedure Iterated Local Search (*time*)
Generate initial solution $P_{initial}$
$P^* = local\ search(P_{initial})$
repeat until time expired
$P' = perturb(P^*)$
$P'^* = localSearch(P')$
if the profit for P'^* *is greater than the profit for* P^* *then*
$P^* = P'^*$

Various options are available when deciding on the end condition for the algorithm and the LSO. In the numerical experiments, we decided to use the time limit as the end condition, which can be set as a parameter. Furthermore, different solution acceptance criteria can be chosen for the PO: starting each perturbation from the best so far, from the current local search result or some other solution found during the algorithm run history. The algorithm runtimes for our problem numerical experiments, as described in Section 6, are very long, therefore, we decided to focus the search as much as possible, always using the best known solution as the perturbation starting point. In the ILS literature, such acceptance criterion is usually called the *best* acceptance criterion (Lourenço et al., 2003).

3.3. Initial solutions

Initial solutions $P_{initial}$ are randomly generated trip pricing tables with each element $P^i_{zw} \in P_{initial}$ in the given interval $[p_{min}; p_{max}]$. They are obtained using a random number generator that generates numbers with approximately uniform distribution in the specified interval. Preliminary tests have shown that the quality of initial solutions varies significantly, depending on the choice of the price interval $[p_{min}; p_{max}]$. The influence of these parameters is analyzed further in Section 6.

3.4. Local search

The LSO used in our approach is explained in the pseudo-code in Algorithm 2. It is a simple method that iteratively increases and then decreases trip pricing table elements, as long as these changes improve profit. The procedure has two parameters: *step* and *time*. The *step* parameter defines the smallest change in price that can be made during the search and the *time* parameter defines the longest allowed search duration. The interval in which the local search can modify the solutions is p_{min} and p_{max}. The price interval should be selected as a reasonable interval for the problem at hand.

The order in which table elements are modified is randomized to encourage the discovery of features for which the order of price changes matters, thus the operator is non-deterministic. For each considered element of the table, the operator first tries to increase the price by adding *step* to the initial price. If the modification causes a better profit, further increases will be performed until p_{max} is reached or price increases are no longer improving the profit (or the time expires). The analogous procedure is followed for price decreases. After the benefits of both increasing and decreasing the price have been examined, the algorithm updates the trip pricing table accordingly so that the new value gives the highest profit gain or retains the old value if price changes caused a profit drop. The subset of the prices to consider changing can vary, but in our implementation, we decided to search through the entire table, i.e. the changing candidate set of prices for the LSO, denoted C_{ls} is a set of all table elements. After all of the elements of C_{ls} have been considered for modification, the operator will start again, but it will visit the elements in a new randomly generated sequence.

The above procedure continues until on entire pass through the table has been done without any improvements being made or until the allowed time has elapsed. By systematically exploring the effect of the price variations and combining the contributions of many small price changes, the LSO can yield significant solution improvements, as shown in the numerical application.

Some additional notation used in Algorithm 2:

$\left(P^i_{zw}\right)_{down}$: Potential new lower price considered by the LSO and calculated by lowering the initial price P^i_{zw}.

$\left(P^i_{zw}\right)_{up}$: Potential new higher price considered by the LSO and calculated by increasing the initial price P^i_{zw}.

Algorithm 2. Pseudo-code of the *local search* operator (LSO).

Procedure local search (**step**, **time**)

*Repeat until **time** expires*
  Initialize random list of table elements C_{ls}
  For each $P^i_{zw} \in C_{ls}$
    $\left(P^i_{zw}\right)_{down} = P^i_{zw}$,
    $\left(P^i_{zw}\right)_{up} = P^i_{zw}$
    *Repeat while profit is increased, $(P^i_{zw})_{down} > P_{min}$ and **time** is not expired*
      $(P^i_{zw})_{down} = (P^i_{zw})_{down} - step$
    *Repeat while profit is increased **and** $(P^i_{zw})_{up} < P_{max}$ and **time** is not expired*
      $(P^i_{zw})_{up} = (P^i_{zw})_{up} + \boldsymbol{step}$
    Update P^i_{zw} to the element of $\{P^i_{zw}, (P^i_{zw})_{up}, (P^i_{zw})_{down}\}$for which the profit is maximal

3.5. Perturbation

The PO, presented in Algorithm 3, introduces random price changes in a small subset of the price table elements. The operator has two parameters: maximum number of elements to change (n) and the maximum allowed change (d). First, n elements from a price table $P(p_{min}, p_{max})$ are randomly selected into the perturbation modification candidate set $C_p \subseteq P(p_{min}, p_{max})$. Then, for each element $P^i_{z,w} \in C_p$, the new price is calculated by adding a random value $\Delta p \in [-d, d]$ to the previous value. The interval, in which the perturbation can change the prices, varies between p_{min}, and p_{max}. If the price after adding Δp is lower than p_{min}, it is updated to p_{min}, and likewise, if it is greater than p_{max}, it is updated to p_{max}.

The set C_p is called the modification candidate set, since there is no guarantee that all of its members will be changed. Due to the definition of the interval from which Δp values are selected, it is possible that for some elements Δp will be zero, leaving the table elements unchanged. This behavior is intentional to ensure greater variability of the perturbation effects on a candidate solution.

It should be noted that *local search* (LSO) and *perturbation* (PO) operators are structurally related in such a way that the local search is unlikely to cancel the effects of perturbation. If *step* is greater than 0.01 €/min, the local search can return to the previous local optimum only if all of the changes caused by the perturbation are multiples of the search *step* value. The probability for such an event to occur drops very quickly as d grows in comparison to *step* and $n > 1$. Nevertheless, finding a balanced perturbation intensity is still very important, to ensure that it is not too strong, as shown in the numerical application in Section 5.

Algorithm 3. Pseudo-code of the *perturbation* operator (PO).

*Procedure **perturb**(**n**, **d**)*

*Initialize set C_p containing **n** random table elements*
For each table element $P^i_{z,w} \in C_p$
  Set Δp to a random value in interval $[-\boldsymbol{d}, \boldsymbol{d}]$
  Modify table element: $P^i_{z,w} = P^i_{z,w} + \Delta p$

4. The case study of Lisbon (Portugal)

The case study used in this work is the municipality of Lisbon, in Portugal. This municipality has been dealing with several mobility problems, including traffic congestion and shortage of parking space associated with the increase in car ownership and the consequent high use of private transport. Public transport has been upgraded; however, it has not been able to reduce the use of private transportation for commuter trips. There is a need to manage mobility in a smart way by, for instance, encouraging transport alternatives such as carsharing.

The data needed to study the trip pricing methodology are: a set of stations, a carsharing trip matrix, a reference price, the price elasticity of carsharing demand, driving travel times between the set of stations, and the system's operating costs. The possible station locations were defined by a grid of squared cells (with sides of length 1000 m) over Lisbon, and associating one location with the center of each cell, which resulted in a set of $K = 75$ possible station locations. This is obviously a simplification, but it serves the purpose of the application. The trip matrix was based on a geo-coded survey updated in 2004 in the Lisbon Metropolitan Area. The survey provides several data, including trip origins and destinations, time of departure, and transport mode used for each trip. Thus, the trip matrix had to be filtered through criteria, such as age of the travelers, trip time, trip distance, time of the day, and transport mode used, so that only the trips that could potentially be served by carsharing were considered, resulting in 1777 trips.

As far as we know, there are no studies in the literature that specifically address the calculation of carsharing price elasticity of demand. Therefore, we decided to use a value of $E = -1.5$, which is the price elasticity of vanpooling demand found by (York and Fabricatore, 2001), because it is the transport mode most similar to carsharing for which information is available. Travel times were computed using the transportation modeling software VISUM (PTV), taking the Lisbon network and the car trip matrix for the entire region, and they are expressed in minutes. We assumed that the carsharing system is available 18 h per day, between 6:00 a.m. and midnight.

To compute the costs related to the vehicles, we take as reference an "average" car mainly driven in the city that costs €20,000 initially, thus yielding the following parameter values:

C_{mv} (cost of maintaining a vehicle): 0.007 euros per minute. This cost was calculated using INTERFILE (INTERFILE, 2012), a tool available on the internet that was developed by a German company, and includes insurance, fees, taxes, fuel, maintenance and wear of the vehicle;

C_v (cost of depreciation per vehicle): 17 euros per day, again calculated with INTERFILE and anticipating 3 years' use in the system. It was also assumed that the company needed to fully finance the purchase of the vehicles at an interest rate of 12% and vehicle residual value of €5000;

C_{mp} (cost of maintaining a parking space): 2 euros per day, this cost is less than the parking fee in a low price area in Lisbon, on the assumption that the city authorities would give support to these types of initiative.

The base carsharing price per minute, $P0$, was considered to be 0.3 euros per minute, which is based on the rates of *car2go* (car2go, 2014). Note that there is no linkage between this price and the demand that is going to be used for the computational experiments, since carsharing is not currently offered in this city.

With these data, the VRPOCS was implemented with a time step of one minute and solved using Xpress 7.7, an optimization tool that uses branch-and-cut algorithms for solving MIP problems (FICO, 2014). The model was first run with no relocation operations, resulting in a daily deficit of €1160.7, which proves the need for balancing strategies. Then the VRPOCS was run with null vehicle relocation cost. This solution produced the relocation balance vector $\in^0_{k_t}$. Time was then divided into 6 intervals for computing the time interval relocation balance ($\omega^0_{k_i}$): 6:00 a.m. to 8:59 a.m., 9:00 a.m. to 11:59 a.m., noon to 2:59 p.m., 3:00 p.m to 5:59 p.m., 6:00 p.m. to 8:59 p.m, and finally from 9:00 p.m. to midnight.

Stations were grouped into 5 zones using the clustering algorithm described previously ($u = 5$), applied with vector $\omega^0_{k_i}$. Five seems to be a reasonable number to capture the different trip patterns between the stations and maintain computation manageability. Results of the clustering algorithm application are presented in Fig. 1, where, for analysis purposes, we numbered each cluster in each interval according to its typical behavior over the day. It is possible to see that the belonging of each station to each of the 5 zones varies with the 6 time interval relocation patterns, that is, a station may belong to a zone in the first time interval and to another zone in the next one. Fig. 1 also shows the number of trips entering (upper right side of the station) and exiting (lower right side of the station) each station according to the demand vector $D0_{k_t j_{t+\delta^t_{kj}}}$ (no relocated vehicles are considered in these numbers).

Stations in zones 1 and 4 are mostly located in the central business district (CBD); however, zone 4 also has several stations located in the periphery at lunch time. Therefore, these have a higher number of trip destinations in the morning against more trip origins in the afternoon, but zone 4 seems to be including lunch time commuters too. These zones generally contain only a few stations over the day. Zone 2 includes stations in the CBD and in the periphery, their number varying greatly over the day. Despite that variation, the trip pattern remains the same with more arrivals than departures throughout the day. It therefore makes sense that zone 2 provides more vehicles than it requests. Zone 3 contains the highest number of stations for most of the time intervals, with special emphasis on the beginning and the end of the day. The stations included in this zone are mainly located on the periphery, but some are also located in the CBD. This mixed behavior makes the difference between relocated vehicle arrivals and relocated vehicle departures not significant. Finally, zone 5 is mostly a mixed zone.

5. Computational experiments

The TPPOCS mathematical model was also implemented using Xpress 7.7 with the same data that was used in the VRPOCS model. The ILS metaheuristic was implemented in Java 1.8 programming language and Xpress Java Application Programming Interface to gain access to the model. All experiments were performed on two identical computers equipped with a 2.4 GHz Intel Core i7-4700HQ processor and 16 GB of RAM and using Java 1.8.0_11-b12 runtime environment under Windows 8.1 operating system.

A single run of the TPPOCS model takes around 30 s and approximately one minute when 8 instances of the model are running simultaneously. Most of the algorithm runtime is therefore spent evaluating the candidate solutions; our benchmarks have shown that, in the best case, only around 430 evaluations could be performed in one hour, using all of the eight logical processor cores in parallel. This fact strongly influenced the tuning process of the algorithm as well as the algorithm design itself. As hinted in Section 4, to obtain good solutions as quickly as possible, strong intensification is performed through detailed local search and the use of the *best* perturbation acceptance policy (Lourenço et al., 2003). The strong

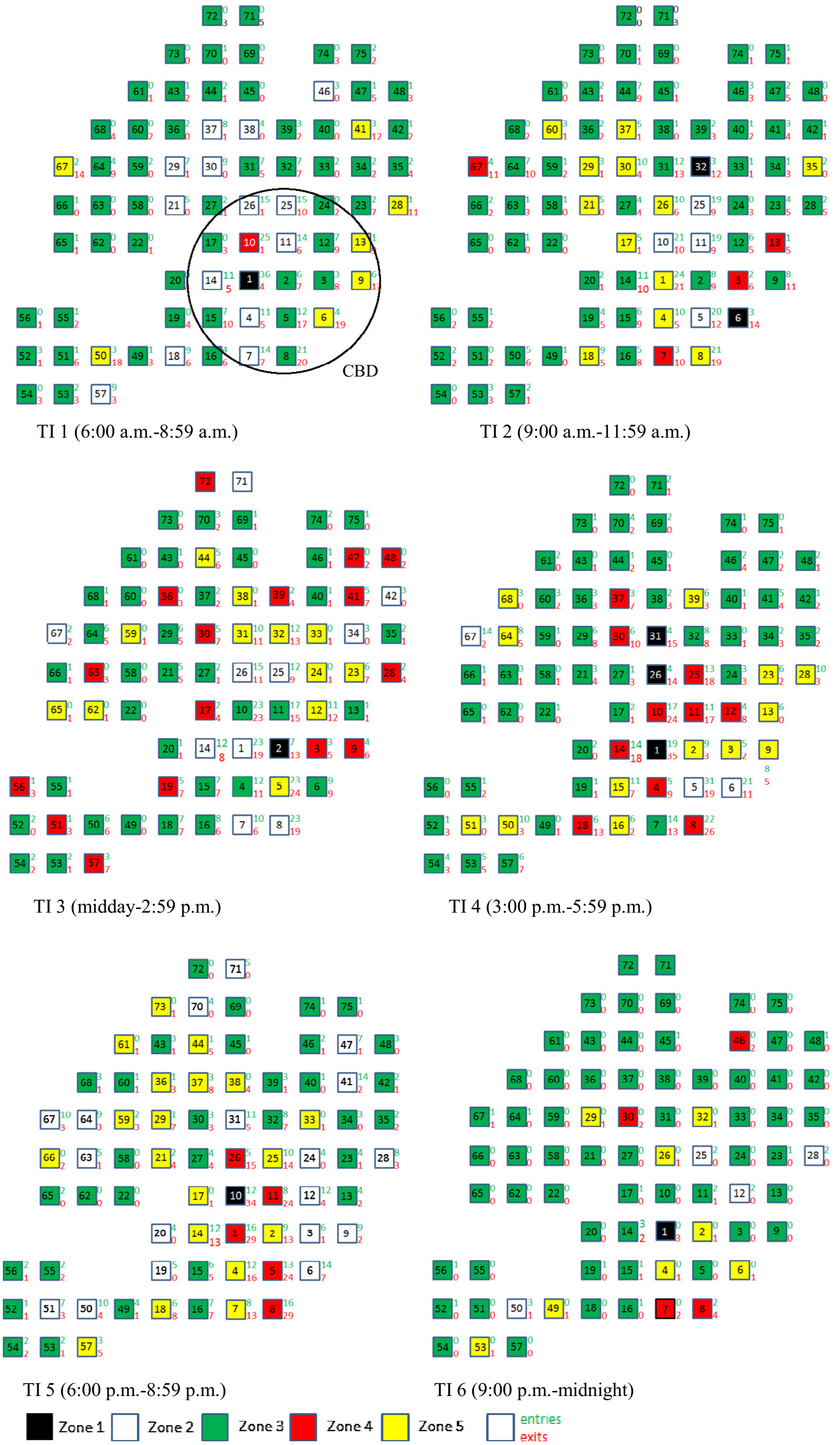

Fig. 1. Stations grouped in zones and number of trip entries and exits at each station in each time interval TI.

intensification is introduced to facilitate the discovery of profit-increasing features as soon as possible. The rest of this section gives an overview of the tuning process and the best results found by the algorithm are presented in the next section.

5.1. Parameter tuning

Metaheuristic methods usually come with a set of parameters that need to be set to specific values. They can significantly influence the metaheuristic performance and, while a quick setup based on the implementer's intuition might work, an experimental evaluation of the influence of the parameters is usually performed to ensure the parameters are adapted to the problem instances to be solved and improve the algorithm results (Birattari, 2005).

The parameter tuning for this problem was done in three stages:

1. Initial solution generator tuning,
2. Local search tuning,
3. Perturbation tuning.

Initial solution generator tuning consisted of determining price bounds p_{min} and p_{max} for the initial solutions. Initial solution tuning rationale is based on the assumption that good initial solutions will enable the local search to find better results more quickly. In total, 105 different intervals were explored: each interval with p_{min} and p_{max} being a multiple of 0.05 €/min in the range [0.00; 0.70] €/min, with 50 solutions generated in each of these intervals. The results proved that the average daily profit for the initial solutions varied greatly depending on the price interval.

The worst profit was measured for the interval [0; 0.05] €/min, (average deficit of 16,714.2 €/day) and the best profit, zero, was achieved for any interval with p_{min} and p_{max} above 0.5 €/min. While at first this might seem like a good result, it should be pointed out that the simulated carsharing demand adapts to price. When unreasonably high prices are applied, the demand drops to zero by force of the elasticity. Zero demand causes the model to shut down the service, interpreted as a zero profit result.

To take this into account, both profit and demand were considered for the initial trip pricing tables. The scatter plot of a subset of explored initial intervals can be seen in Fig. 2, with average profit on the x-axis and average demand on the y-axis. The Pareto non-dominated set of (*profit*, *demand*) pairs, indicated by gray dots on the chart, was selected as a set of candidate intervals (Powell, 1964; Powell, 1977). The highlighted interval [0.35; 0.40] €/min has the lowest average deficit (32.48 €/day) while also retaining high average demand (1579 trips, which corresponds to 89% of the demand with the reference price). Configurations with higher profit do exist, but for them, the demand drops to nearly zero, as can be seen in the example of the interval [0.45; 0.65] €/min, indistinguishable from the zero demand interval [0.5; 0.7] €/min. Having near-zero demand in the system is clearly not the desired result. Therefore, the values of $p_{min} = 0.35$ €/min and $p_{max} = 0.40$ €/min are selected as the initial randomly generated solutions.

The local search tuning consisted of determining the best search *step* parameter. Five initial solutions with price intervals set up according to the values given above were randomly selected and for each of them, local search with *step* equal to €0.01, €0.02, €0.05 and €0.10 was applied. The experiment was repeated five times, resulting in 100 test runs with each local search being limited to run for 4 h. It is assumed that a better functioning local search will provide good results faster in the

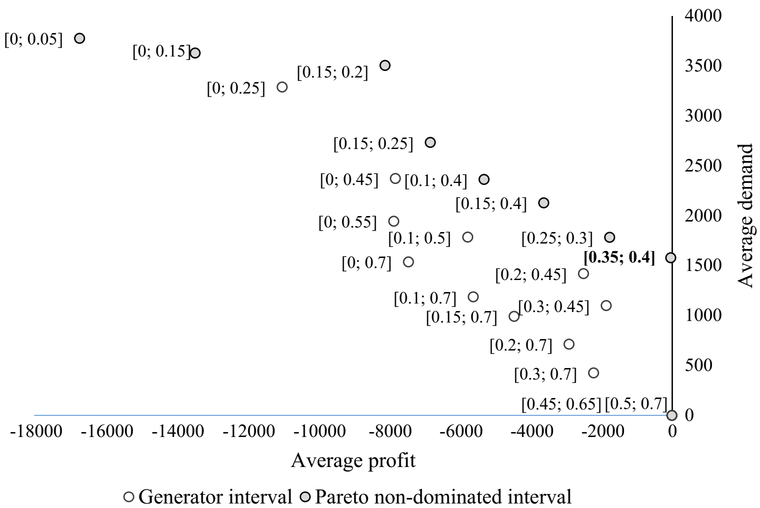

Fig. 2. Initial solution generator configurations.

environment of the ILS. The results of the experiments are given in Table 2, showing average, median, minimum, maximum and sample standard deviation of the profits obtained with local search results. As can be seen in this table, the best results were in general achieved using a *step* of €0.02. The best values found for each experiment are highlighted in bold.

Perturbation tuning consisted of trying to identify which pair of changed set size and intensity (n, d) works best with the LSO configured as in the previous stage. A total of 8 different configurations were run 5 times for 12 h, to determine which perturbation parameters best fit the balance of diversification and intensification as described in Section 4. The results of the experiments are shown in Table 3, with average, median, minimum, maximum and sample standard deviation of the profits obtained with different perturbation settings. As the preliminary tests have shown, the model is very sensitive to price changes. Best average profits are achieved with low perturbation intensities $(n = 2, d = 0.02)$ and $(n = 5, d = 0.02)$, which have almost equal average profit. If perturbation is more pronounced, average profits fall. This can be explained by the fact that the local search is unable to find good solutions before another intensive round of perturbation reduces the profit of the current local optimum.

It is interesting to note that higher perturbation settings can yield very good results. In fact, the best result found during our experiments came with a rather high perturbation setting $(n = 10, d = 0.02)$. Appropriate perturbation settings can be chosen based on the planned algorithm run duration. For short runs, it is important to keep both n and d low to ensure search intensification. For longer runs, however, setting the perturbation to higher levels might be beneficial. With long algorithm runs (12 h or longer in our preliminary experiments) the search progress tends to become stuck in a very good solution that local search cannot further improve. Higher perturbation rates might help the algorithm to change the high profit solutions sufficiently to diversify the search around very high profit solutions, thereby preventing stagnation.

To conclude the algorithm parameter analysis, in Table 4, we provide recommended metaheuristic parameters for solving the TPPOCS, using Lisbon as a case study. The time limits in the table are valid for equipment with similar performance to our experimental setup (around 1440 model evaluations available in 12 h).

5.2. Performance in various problem instances

While parameter tuning is essential to achieve good performance, it is also a lengthy procedure. However, it might not be necessary unless problem features change significantly. We performed a series of experiments on various problem instances, based on our case study data with different elasticity and with different number of clusters.

Table 2
Local search parameter exploration.

Step (€)	Profit (€/day)				
	Average	Median	Min	Max	Sample standard deviation
0.01	1164.8	1224.5	732.7	1561.0	206.7
0.02	**1307.7**	**1338.0**	763.2	**1727.1**	245.4
0.05	1262.0	1302.7	**940.4**	1455.8	140.1
0.10	1164.2	1118.1	830.8	1589.4	193.7

Table 3
Perturbation parameter exploration.

n	*d* (€)	Profit (€/day)				
		Average	Median	Min	Max	Sample standard deviation
2	0.02	**1782.7**	**1827.1**	**1624.7**	1846.7	91.4
2	0.05	1660.5	1730.1	1190.1	1905.5	273.5
5	0.02	1767.3	1796.8	1554.6	1954.5	172.6
5	0.05	1628.8	1720.1	1349.8	1790.8	185.3
10	0.02	1674.2	1783.4	1187.9	**2068.1**	357.7
10	0.05	1560.5	1561.6	1427.3	1721.4	110.1
20	0.02	1696.5	1761.2	1456.3	1885.8	172.2
20	0.05	1618.3	1646.4	1469.6	1743.3	124.7

Table 4
Recommended metaheuristic parameters.

	Initial solution		*step* (€)	*n*	*d*
	P_{min} (€)	P_{max} (€)			
Short runs (⩽12 h)	0.35	0.40	0.02	2	0.02
Long runs (>12 h)	0.35	0.40	0.02	10	0.02

In all cases, the set of parameters for the short runs given in Table 4 is used as a reference configuration and the results with this set of parameters are referred to as the reference results. As described in Sub-section 5.1, the best profit ($\theta = 2068$ €/day) is generated using higher perturbation intensity than that used in the reference parameter set. However, the reference set has the highest median and average results. We find them a better performance indicator, since the maximum can be a single outlier in an otherwise suboptimal result set. Therefore, the reference set of parameters in Tables 5–7 has a slightly lower *max*. value ($\theta = 1846.7$ €/day) than the best found in all experiments.

5.2.1. Influence of the price elasticity of the demand

As shown in Table 5, the elasticity influences the results achieved by the algorithm when reference parameters are used. The metaheuristic algorithm was run 5 times for each of the problem instances and Table 5 shows the statistics of best solutions found during each of the runs. Very good results can be obtained when the demand does not significantly depend on the price, as can be seen for the elasticity E=-0.8 (best profit found is 3341.4 €/day). This can be explained by the economics fact that low elasticity enables service providers to raise prices without rejecting customers. Price tables for the best solution therefore have much higher values than the best found for the reference elasticity $E = -1.5$, the prices are in the interval [0.35; 0.59] €/min and the average price is 0.44 €/min. With such high prices, it is clear that the algorithm detected that significant price increases are a good way to improve profit in this, relatively inelastic, case.

For the high elasticity cases, the metaheuristic is less able to increase profits to high values. Price increase causes deeper falls in the demand without improving the profit much, and each decision has more influence on the entire system. The lowest profits were registered for the most elastic problem we considered, where the worst solution has a daily profit of €95.0 and the average is 497.2 €/day. From Table 5, it is clear that quality of the results declines as the demand is more elastic. Nevertheless, the algorithm was able to find profitable solutions in each of the runs we considered.

5.2.2. Influence of the number of clusters

Number of clusters is a very important question that needs to be addressed carefully in order to provide good solutions. To investigate its influence, we conducted two series of experiments with various cluster numbers: (*i*) using the reference metaheuristic parameters, and (*ii*) tuning the metaheuristic parameters for each cluster number. This provides a more objective assessment of the influence that the number of clusters has on the complete methodology.

As shown in Table 6, the number of clusters influences the performance of the objective function when the reference parameters are used. The table shows average, median, worst, best and the standard deviation of the profit of the best solutions found in five runs of the algorithm, all of them limited to 12 h. The best results are achieved with five clusters. For only

Table 5
Algorithm performance with different price elasticity of demand (E) and no additional tuning.

E	Profit (€/day)				
	Average	Median	Min	Max	Sample standard deviation
−**0.8**	**3063.83**	**3179.56**	**2641.99**	**3341.38**	273.93
−1.0	2645.67	2683.66	2485.71	2711.37	92.46
−1.2	2392.43	2412.07	2271.82	2507.65	88.10
−1.5 (*reference*)	1782.74	1827.14	1624.69	1846.69	91.36
−1.7	966.18	1256.72	383.84	1425.63	521.48
−2.0	497.16	278.13	94.97	1000.81	398.30

Table 6
Algorithm performance with different zone number and no additional tuning.

Zone number	Profit (€/day)				
	Average	Median	Min	Max	Sample standard deviation
2	823.97	943.84	265.88	999.30	312.94
5 (***reference***)	**1782.74**	**1827.14**	**1624.69**	**1846.69**	91.36
7	1497.14	1520.42	1199.19	1716.82	199.51

Table 7
Algorithm performance after tuning for different zone numbers.

Zone number	Profit (€/day)				
	Average	Median	Min	Max	Sample standard deviation
2	1067.75	1081.08	1028.26	1090.72	28.20
5 (***reference***)	**1782.74**	**1827.14**	**1624.69**	**1846.69**	91.36
7	1505.83	1533.99	1218.02	1701.87	178.96

two clusters, the performance is notably bad. It is better with seven, but not as good as for five clusters. These results show that even with a different number of clusters, the metaheuristic is able to reach good results. But if the top performance is desired with given clusters, tuning can improve them, as demonstrated in the next experiment.

To demonstrate the influence of clustering on the entire methodology, we conducted a series of experiments during which the metaheuristic was specifically tuned to adapt to the current number of clusters. The tuning procedure followed was exactly the same as for 5 clusters in each of the experiments. By varying only the number of clusters in the entire methodology, it was possible to isolate the influence this has on the results. After tuning the algorithm to adapt it to the difference in the cluster number, the performance improved, as expected. The performance of the tuned metaheuristic is demonstrated in Table 7, with the performance indicators showing that the best results are achieved with five clusters. In the two cluster example, the algorithm reaches the local optimum relatively quickly and even restarting from different solutions makes little difference, probably due to the fact that a lot of valuable nuances that describe the trips between stations are lost in the coarse grained clustering. Conversely, in the seven clusters example, the pricing tables have 294 elements compared with 150 for the reference case of five clusters. This increase in the price table size impairs the search procedure's ability to produce local optima quickly enough, which leads to inferior results. The differences in the results show that even with the algorithm separately tuned to each instance, the number of clusters can influence the performance. Some practitioner guidelines to choosing the appropriate cluster number are therefore discussed in the Sub-section 5.2.3.

To conclude the experimental evaluation, we have demonstrated that the metaheuristic is able to work well with various different problem types and yield profitable results in all of the modified problems, even when it is tuned to work best with a different elasticity or cluster number. Still, we should remember that the highest performing set of parameters is problem instance specific. Therefore, in a different city, with different trip patterns or other notable changes, the set of good metaheuristic parameters may differ.

5.2.3. Clustering and time interval detail level: precision vs. simplification

When deciding on the number of clusters and time intervals for modeling a new city, they should be sufficiently large to faithfully model the mobility patterns and classify the stations in enough detail, but not so big as to diminish metaheuristic performance. For example, if a city has rush hours in the morning (around 8:00 a.m.) and in the afternoon (around 4:00 p.m.), dividing the day in two intervals (7:00 a.m.–7:00 p.m.) and (7:00 p.m.–7:00 a.m.) is clearly a bad choice. Because both rush hours happen in a single time interval (the first one), this oversimplification will cause the system to treat them both the same, despite the fact that they typically move in opposite directions. The insufficient detail level of only two time intervals and a bad choice of the interval start will therefore be likely to prevent the system from efficiently taking advantage of differences in mobility patterns in the morning and afternoon rush hours. Similar reasoning can be given for the cluster detail level.

The number of clusters and time intervals only has to be enough to allow the algorithm to do detailed decisions that take advantage of the user behavior in the best possible way. Despite the greater precision it brings, making the number of clusters too high may not be beneficial. As mentioned in Sub-section 2.3, the number of possible candidate solutions increases exponentially the higher the number of clusters and time intervals. Too many clusters and intervals cause the metaheuristic to converge rather slowly due to the increased computational complexity arising from the combinatorial explosion of the search space size. The number of clusters and time divisions can be a part of the metaheuristic tuning as well, if the initial results with the predefined values are not satisfactory. The bound analysis formulas and models provided in sub-Sub-section 4.2 can be highly useful guidelines to arriving at educated decisions while implementing our methodology. If the solutions found by the metaheuristic algorithm are far from the optimum, or even lower than the lower optimum bound, this can be a good indicator that a tuning procedure, although lengthy, might be beneficial to the system. Likewise, if the profit is closer to the upper optimum bound (which is clustering independent), it is a sign that the metaheuristic is approaching the upper limit of its capabilities. In these cases, further tuning and increases in the cluster number will almost certainly produce little or no improvement.

6. Results

The best trip pricing table found in our experiments, denoted P_{best}, is presented in Table 8. Using this pricing table, the system is able to achieve a profit of 2068.1 €/day. Compared to the loss of 1160.7 €/day resulting from not implementing any balancing strategy and having to satisfy all the reference demand (1777 trips) for the reference price, it is clear that variable pricing can lead to significant profit increases. For the best trip pricing table found, satisfied demand is 1471 trips per day, which is a loss of 306 trips in relation to the reference demand (17.7% demand reduction). We should note, however, that this demand is not rejected per se; it indicates that some travelers will find the price too high to use the carsharing service. The average price charged is 0.39 €/min, with all the prices being in the interval [0.35; 0.46] €/min, that is, all prices charged are higher than the reference carsharing price ($P0$), which is 0.30 €/min.

Fig. 3 presents the difference between the vehicle entries and exits in each station before applying trip pricing and without any relocations, and the difference between entries and exits with the trip pricing. Before applying trip pricing, time intervals 1 and 5 have considerable entry-exit imbalance, since they correspond mostly to the morning and afternoon rush hours. Analyzing Fig. 3, it can be seen that in these periods of the day, the effect of trip pricing is notable for all zones'

Table 8
Best found trip prices for each origin–destination pair of zones and time interval.

TI 1 (6:00 a.m.–8:59 a.m.)						TI 2 (9:00 a.m.–11:59 a.m.)						TI 3 (midday–2:59p.m.)					
Zones	1	2	3	4	5	Zones	1	2	3	4	5	Zones	1	2	3	4	5
1	0.36	0.44	0.38	0.40	0.38	1	0.35	0.43	0.38	0.41	0.40	1	0.40	0.39	0.39	0.40	0.41
2	0.38	0.39	0.39	0.38	0.39	2	0.39	0.39	0.39	0.39	0.39	2	0.38	0.39	0.38	0.39	0.39
3	0.46	0.37	0.38	0.44	0.38	3	0.38	0.38	0.39	0.38	0.39	3	0.39	0.39	0.39	0.39	0.39
4	0.35	0.41	0.36	0.38	0.39	4	0.39	0.40	0.39	0.41	0.40	4	0.38	0.40	0.38	0.39	0.39
5	0.39	0.45	0.35	0.41	0.39	5	0.38	0.39	0.38	0.36	0.38	5	0.39	0.39	0.39	0.39	0.39
TI 4 (3:00p.m.–5:59p.m.)						TI 5 (6:00p.m.–8:59p.m.)						TI 6 (9:00p.m.–midnight)					
1	0.39	0.43	0.39	0.39	0.38	1	0.39	0.38	0.39	0.38	0.45	1	0.39	0.36	0.38	0.38	0.36
2	0.38	0.38	0.38	0.38	0.39	2	0.39	0.38	0.38	0.38	0.39	2	0.40	0.39	0.40	0.35	0.40
3	0.39	0.39	0.38	0.38	0.39	3	0.36	0.39	0.40	0.38	0.38	3	0.38	0.36	0.39	0.38	0.39
4	0.39	0.39	0.39	0.37	0.46	4	0.38	0.40	0.39	0.39	0.38	4	0.35	0.35	0.38	0.36	0.36
5	0.39	0.39	0.38	0.38	0.35	5	0.39	0.38	0.41	0.38	0.39	5	0.39	0.38	0.40	0.38	0.38

The OD pairs of zones in which there is a decrease in the demand due to the increase of price are highlighted with gray.

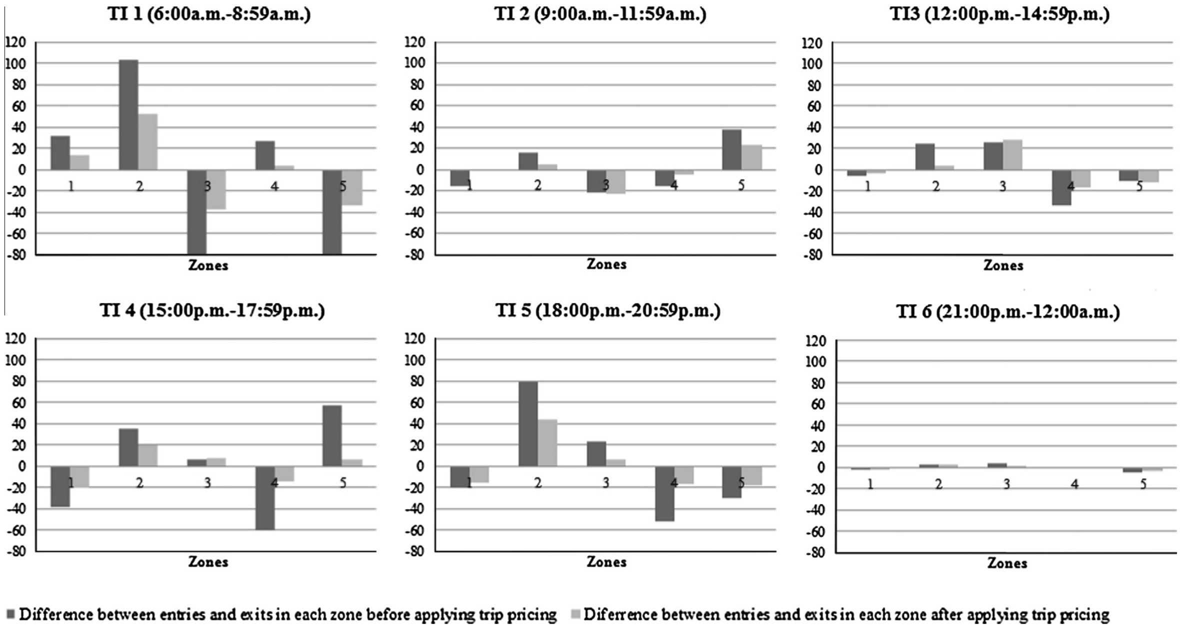

Fig. 3. Difference between trip entries and exits for each zone and time interval (TI), before and after applying trip pricing.

balance, whilst in time intervals 2, 3 and 4, this effect is not so obvious. Furthermore, in these three intervals, zone 3 seems unchanged. This may be because this zone encompasses more stations, which implies more variability in trip patterns.

Most of the OD pairs of zones that have a significant fall in demand correspond to a price of 0.40 €/min or higher. We should mention that the elasticity is being applied to the unit price per trip and not to the total price of a trip; thus, we are not considering that longer trips may give different results from shorter ones.

With the results it is possible to do a more detailed analysis of what happens in each time interval for each zone and trip direction. In the early morning (6:00 a.m. to 8:59 a.m.), there are mainly demand reductions through price increases for the trips that depart from zones 3 and 5, which are located in the periphery and so account for many more trip origins than destinations in this period. Demand is also reduced for trips that arrive at zone 2, which is located in the center and therefore has many more trips arriving (about 70% of the trips) than beginning there. This is because people tend to travel from residential areas to the CBD in the morning. Moreover, this is the interval with the highest prices for all OD pairs of zones, which confirms that it is the most imbalanced one.

Periods 2 and 3 (9:00 a.m. to 11:59 a.m and midday to 14:59 p.m, respectively) are mostly intermediate periods when most of the prices charged are the same as the average price charged for the entire day (0.39 €/min) or lower. However, despite there not being much difference between number of departures and arrivals, when there is a difference, the prices charged act to reduce the demand in the most imbalanced directions.

Time interval 4 (15:00 p.m. to 17:59 p.m) is also an intermediate, more balanced time interval. Its proximity to the afternoon peak hour is noticeable as there are already some return home trips. The prices charged in this time interval also reflect this fact by being higher than the average for some OD pairs of zones. The model acts to lower the demand for departures from zones 1 and 4 located in the center, since these outbound trips are more than 50% of the total trips for these two zones, and it decreases the demand that arrives at zone 5, located on the periphery. Zone 5 has more trip destinations than origins (about 68%/32%). For the OD pair 4–5, the price charged is 0.46 €/min, which is the highest price charged in the pricing table.

The afternoon peak period (18:00 p.m. to 20:59 p.m) shows a greater imbalance of vehicles across all the stations. Therefore, the demand reductions in response to the price increase are more pronounced in this period. For example, they exist for both directions in zone 4. In this period, zone 4, located in the city center, shows a higher number of trip origins than trip destinations due to work-home trips. Thus, a decrease in the demand for this zone as destination was not expected. However, trip arrivals are also lower, as expected, and the effect of reductions in both ways results in a more balanced zone at a scale that is manageable by the whole network. This time interval sets a price of 0.45 €/min for OD pair 1–5 (the second highest price), which was expected, since there is a very imbalanced movement at this time of the day with trips mostly from the center to the periphery.

Finally, the end of the day, which corresponds to time interval 6 (21:00 p.m. to midnight) has few trips (only 11). For this reason, there is no need to reduce the demand significantly and, at the same time, price reductions for demand increase are apparently not beneficial either.

The improvement in the profitability of the company is not only due to the decreased demand for some OD pairs of zones and time intervals, it is also due to the price increase itself in many OD pairs where, while it is not enough to produce an expected demand reduction, it is sufficient to have an impact on increasing the profits. This occurs even though we have considered in the case study that demand is elastic to price variations, elasticity is greater than 1 (absolute value), which should point to a reduction in profit from price increase in a linear model. The special and complex interdependence of supply and demand in carsharing systems leads to a system that is beneficial when run for a lower number of trips, yet one that is more balanced.

Zoning that was determined by computing a theoretical desired relocation vector is able to divide the stations into sets for which the price variations yield a higher profit. Even though using the metaheuristic does not guarantee the optimal solution will be found, we can still demonstrate through its application to the case study that an increase in prices can actually lead to a higher profit; a solution that not only avoids losses (system closure will generate 0 profit) but that is able to generate positive and significant profits. We may conclude from the global results presented in Table 9 that the balance of vehicles and profit are directly related, because a more balanced system results in higher revenue, requiring fewer vehicles and fewer parking spaces, which means lower operating costs.

It is also interesting to evaluate the metaheuristic performance given the bounds calculated as proposed in Sub-section 2.4. The best profit solution found with the metaheuristic algorithm and the theoretical profit bounds are compared in the graph in Fig. 4. The worst possible financial outcome in our case study, with free rides for all and greatest possible demand, is 24534.8 €/day. The lowest bound of the optimal profit, calculated as the profit in a perfectly balanced solution is 316.0 €/day, whereas the upper bound is 12027 €/day. The metaheuristic algorithm produced a solution with a profit almost 7 times higher than the lower bound. This result also demonstrates that just balancing the system is not enough to achieve high profits. The result found by the metaheuristic is not too far from the optimum, especially if we consider that the upper bound is a highly conservative limit that can be achieved only in theory, when the demand is not rounded to the unit and neither vehicle ownership costs nor parking space costs are considered.

Table 9
Global results with and without trip pricing.

	Profit (€/day)	Revenue related to the trips (€/day)	Costs of vehicle maintenance (€/day)	Costs of vehicle depreciation (€/day)	Costs of parking spaces maintenance (€/day)	Satisfied demand	Fleet of vehicles	Number of parking spaces
No balancing strategy	−1160.7	7113.3	166.0	6630	1478	1777	390	739
Trip pricing	2068.1	7576.4	138.3	4352	1018	1471	256	509

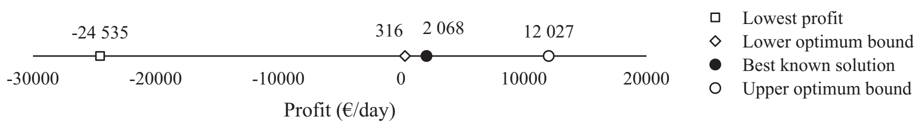

Fig. 4. Theoretical profit bounds compared to the best solution found by the algorithm.

7. Conclusions

There are two main types of carsharing operation models: round-trip carsharing, in which the users have to pick up and return the car to the same station, and the one-way carsharing that allows users to pick up a vehicle at one station and drop it off at another. The latter has been associated with more trip purposes than round-trip carsharing – for example, one way carsharing can also be used for commuting (Balac and Ciari, 2014). However, it also brings up the problem of vehicle stock imbalance and a need to find efficient ways to balance vehicle stocks across stations. Several ways to improve this balance have been proposed in the literature, and some empirical studies have suggested the use of variable trip pricing (Mitchell et al., 2010). This method consists of varying the price charged to the clients based on the stock of vehicles at the origin and destination stations. Though never proven, the theory is that by changing demand through pricing, carsharing systems could yield higher profits.

In this paper, we have proposed a model that considers demand as a function of price and searches for the prices that maximize the profit of the daily operation of a one-way carsharing company. Because this model is non-linear and the objective function is non-concave, we use ILS as a metaheuristic to solve the problem (Stützle and Hoos, 1999; Lourenço et al., 2001; Lourenço et al., 2003; Luke, 2013). For setting the prices, stations were grouped into zones and time was divided into time intervals. Trip prices therefore varied between each OD pair of zones according to the time interval in which the trip begins. This methodological approach was numerically tested for the case study of Lisbon, in Portugal.

The case study application shows that using price variation to balance vehicle stocks across one-way carsharing stations works satisfactorily. When no vehicle balancing mechanism is applied, the carsharing company has a deficit of 1160.7 €/day. In a perfectly balanced solution, a profit of 316.0 €/day is achieved. Using the trip pricing metaheuristic approach, the profit for the best price combination found through the use of the ILS is 2068.1 €/day. This is an increase of 3228.8 €/day over the case of no balancing mechanism in a system that has 75 stations and serves 1471 trips. We demonstrate that system balancing has a very important role in reducing the costs and increasing profitability of carsharing systems. Our perfectly balanced solution has a higher profit than the imbalanced one.

Another important aspect of this work is that it shows that balancing alone is not enough to achieve optimal profits. Using a metaheuristic algorithm to optimize profits, we were able to find solutions offering more than six times higher profit than a perfectly balanced solution. While the best solution found also has significantly decreased imbalance, it is not zero.

It is also relevant to note that the prices charged to the clients for every OD pair of zones increased in comparison to the reference price, which leads to lower demand. However, the increase in price happens through a generalized reduction in the imbalanced demand served by carsharing. The results show that in most cases the OD pairs of zones that have price increases, and therefore a decrease in demand, are the ones with a greater difference between trip origins and trip destinations which demonstrates that the solution algorithm (metaheuristic) is able to capture the essential behavior of the system related to the trip imbalance across zones and time intervals and improve its performance. Additionally it is quite interesting to observe that even though a notable elastic behavior of demand toward price (-1.5) was introduced in the model the general increase of prices produces a higher profit, which in a simpler case of one demand for one price is not intuitive, but in this case is the result of a complex system with multiple feedbacks that characterizes carsharing.

The main conclusion that is drawn from this study is that trip pricing can be considered an effective method to improve the profitability of one-way carsharing systems. Concerning the generalizability of the method, we have shown that it is possible to successfully apply it under variability such as changes in the way stations are divided into clusters, or changes in consumer habits that impact price elasticity over time. Therefore, the methodology is robust enough to work reasonably, even if some of the real world parameters change. Additionally, the metaheuristic is a generic optimization tool that could be used with any other traffic simulator that is able to estimate profit based on a variation in pricing, and is not restricted to the elasticity model we use in our case study. This indicates that our methodology could be applied in various other environments. However, it must be noted that the solution algorithm computation time is dependent on the problem dimension.

The devised bound analysis provides a way to estimate the upper and lower bounds of the optimum that can be employed by practitioners in other cities to give valuable feedback on their current progress and help guide decisions on future applications.

Regarding further developments, we suggest enhancing the solution method proposed. Studying different principles to determine the zones could also be relevant for profit maximization. It would be interesting, too, to investigate the efficiency of pricing as a balancing strategy when combined with other factors such as station-location selection and relocation operations. Moreover, the computation of the demand variation with price, which was performed in a very simple way in this study, should also be improved because it is a key aspect in the realism of the results.

Acknowledgements

The research on which this paper is based was carried out within the framework of a PhD thesis of the MIT Portugal Program. The authors wish to thank Fundação para a Ciência e a Tecnologia (FCT) of Portugal for financing this PhD project through scholarship SFRH/BD/51328/2010. The work was also supported by the InnoVshare project, also financed by FCT (Grant PTDC/ECM-TRA/0528/2012). Special thanks go to Prof. Cynthia Barnhart from the MIT for hosting the main author of the paper at that Institute.

References

Atasoy, B., Salani, M., Bierlaire, M., 2014. An integrated airline scheduling, fleeting, and pricing model for a monopolized market. Computer-Aided Civil and Infrastructure Engineering 29 (2), 76–90.

Balac, M., Ciari, F., 2014. Modelling station-based carsharing in Switzerland. In: 14th Swiss Transport Research Conference, 14–16 May 2014, Monte Verità/Ascona, Switzerland.

Barth, M., Han, J., Todd, M., 2001. Performance evaluation of a multi-station shared vehicle system. In: Proceedings of 2001 IEEE Intelligent Transportation Systems Conference, 25–29 August 2001, Oakland, CA, USA.

Barth, M., Shaheen, S., 2002. Shared-use vehicle systems: framework for classifying carsharing, station cars, and combined approaches. Transportation Research Record: Journal of the Transportation Research Board 1791, 105–112.

Barth, M., Todd, M., 1999. Simulation model performance analysis of a multiple station shared vehicle system. Transportation Research Part C 7 (4), 237–259.

Barth, M., Todd, M., Xue, L., 2004. User-based vehicle relocation techniques for multiple-station shared-use vehicle systems. In: 83rd Annual Meeting of the Transportation Research Board, 11–15 January 2004, Washington, D.C., USA.

Birattari, M., 2005. Tuning Metaheuristics – A Machine Learning Perspective. IOS Press, Incorporated, Amsterdam, Netherlands.

Boyaci, B., Zografos, K., Geroliminis, N., 2015. An optimization framework for the development of efficient one-way car-sharing systems. European Journal of Operational Research 240 (3), 718–733.

Bordley, R., 1994. An overlapping choice set model of automotive price elasticities. Transportation Research Part B 28 (6), 401–408.

Bussieck, M., Vigerske, S., 2014. MINLP solver software. GAMS Development Corporation, car2go. <http://www.car2go.com/> (Accessed 16.08.14).

Carlier, J., 1982. The one-machine sequencing problem. European Journal of Operational Research 11 (1), 42–47.

Celsor, C., Millard-Ball, A., 2007. Where does carsharing work?: using geographic information systems to assess market potential. Transportation Research Record: Journal of the Transportation Research Board 1992, 61–69.

Chung, B., Yao, T., Friesz, T., Liu, H., 2012. Dynamic congestion pricing with demand uncertainty: a robust optimization approach. Transportation Research Part B 46 (10), 1504–1518.

Ciari, F., Bock, B., Balmer, M., 2014. Modeling station-based and free-floating carsharing demand: a test case study for Berlin, Germany. Emerging and Innovative Public Transport and Technologies. CD-ROM. Transportation Research Board of the National Academies, Washington, D.C., USA.

Correia, G., Antunes, A., 2012. Optimization approach to depot location and trip selection in one-way carsharing systems. Transportation Research Part E 48 (1), 233–247.

Costain, C., Ardron, C., Habib, K., 2012. Synopsis of users' behaviour of a carsharing program: a case study in Toronto. Transportation Research Part A 46 (3), 421–434.

de Dios Ortúzar, J., Willumsen, L.G., 2011. Modelling Transport. John Wiley & Sons, Chichester, United Kingdom.

Fan, W., Machemehl, R., Lownes, N., 2008. Carsharing: dynamic decision-making problem for vehicle allocation. Transportation Research Record: Journal of the Transportation Research Board 2063, 97–104.

Febbraro, A., Sacco, N., Saeednia, M., 2012. One-way carsharing: solving the relocation problem. In: 91st Annual Meeting of the Transportation Research Board, Washington, D.C., USA, pp. 113–120.

FICO, 2014. Getting Started with Xpress, Release 7.7. Fair Isaac Corporation, Leamington Spa, UK.

Firnkorn, J., Müller, M., 2011. What will be the environmental effects of new free-floating carsharing systems? The case of car2go in Ulm. Ecological Economics 70 (8), 1519–1528.

Fouquet, R., 2012. Trends in income and price elasticities of transport demand (1850–2010). Energy Policy 50, 62–71.

Hu, P., Reuscher, T., 2001. 2001 National Household Travel Survey: Summary of Travel Trends. U.S. Department of Transportation.

INTERFILE, <http://www.excel.interfile.de/Autokostenrechner/autokostenrechner.htm> (Accessed 10.06.12).

Ji, Y., Geroliminis, N., 2012. On the spatial partitioning of urban transportation networks. Transportation Research Part B 46 (10), 1639–1656.

Joksimovic, D., Bliemer, M., Bovy, P., 2006. Optimal toll design problem in dynamic traffic networks with joint route and departure time choice. Transportation Research Record: Journal of the Transportation Research Board 1923, 61–72.

Jorge, D., Correia, G., 2013. Carsharing systems demand estimation and defined operations: a literature review. European Journal of Transport and Infrastructure Research 13 (3), 201–220.

Jorge, D., Correia, G., Barnhart, C., 2014. Comparing optimal relocation operations with simulated relocation policies in one-way carsharing systems. IEEE Transactions on Intelligent Transportation Systems 15 (4), 1667–1675.

Jorge, D., Barnhart, C., Correia, G., 2015. Assessing the viability of enabling a round-trip carsharing system to accept one-way trips: application to Logan Airport in Boston. Transportation Research Part C: Emerging Technologies 56, 359–372.

Kaspi, M., Raviv, T., Tzur, M., 2014. Parking reservation policies in one-way vehicle sharing systems. Transportation Research Part B 62, 35–50.

Katayama, K., Narihisa, H., 1999. Iterated local search approach using genetic transformation to the traveling salesman problem. In: Proceedings of the 1st Conference on Genetic and Evolutionary Computation, 13–17 July 1999, Orlando, Florida, USA, pp. 321–328.

Kek, A., Cheu, R., Chor, M., 2006. Relocation simulation model for multiple-station shared-use vehicle systems. Transportation Research Record: Journal of the Transportation Research Board 1986, 81–88.

Kek, A., Cheu, R., Meng, Q., Fung, C., 2009. A decision support system for vehicle relocation operations in carsharing systems. Transportation Research Part E 45 (1), 149–158.

Krumke, S., Quilliot, A., Wagler, A., Wegener, J., 2013. Models and algorithms for carsharing systems and related problems. Electronic Notes in Discrete Mathematics 44, 201–206.

Litman, T., 2000. Evaluating carsharing benefits. Transportation Research Record: Journal of the Transportation Research Board 1702, 31–35.

Lopes, M., Martínez, M., Correia, G., Moura, F., 2015. Insights into carsharing demand dynamics: outputs of an agent-based model application to Lisbon, Portugal. In: 94th Annual Meeting of the Transportation Research Board, Washington, D.C., USA.

Lou, Y., Yin, Y., Laval, J., 2011. Optimal dynamic pricing strategies for high-occupancy/toll lanes. Transportation Research Part C 19 (1), 64–74.

Lourenço, H., Martin, O., Stützle, T., 2001. A beginner's introduction to iterated local search. In: Proceedings of the Fourth Metaheuristics Conference, 16–20 July 2001, Porto, Portugal, vol. 2, pp. 1–6.

Lourenço, H., Martin, O., Stützle, T., 2003. Iterated local search. In: Glover, F., Kochenberger, G. (Eds.), Handbook of Metaheuristics. Kluwer Academic, Norwell, pp. 321–352.

Luke, S., 2013. Essentials of Metaheuristics, Lulu, second edition, <http://cs.gmu.edu/sean/book/metaheuristics/>.

Martin, E., Shaheen, S., 2011. Greenhouse gas emission impacts of carsharing in North America. IEEE Transactions on Intelligent Transportation Systems 12 (4), 1074–1086.

Martin, O., Otto, S., 1995. Partitioning of unstructured meshes for load balancing. Concurrency: Practice and Experience 7 (4), 303–314.

Mitchell, W., Borroni-Bird, C., Burns, L., 2010. Reinventing the Automobile: Personal Urban Mobility for the 21st Century. MIT Press, Cambridge, MA, USA, pp. 130–155.

Nair, R., Miller-Hooks, E., 2011. Fleet management for vehicle sharing operations. Transportation Science 45 (4), 105–112.

Nourinejad, M., Roorda, M., 2014. A dynamic carsharing decision support system. Transportation Research Part E 66, 36–50.

Pfrommer, J., Warrington, J., Schildbach, G., Morari, M., 2014. Dynamic vehicle redistribution and online price incentives in shared mobility systems. IEEE Transactions on Intelligent Transportation Systems 15 (4), 1567–1578.

Powell, M., 1964. An efficient method for finding the minimum of a function of several variables without calculating derivatives. The Computer Journal 7 (2), 155–162.

Powell, M., 1977. Restart procedures for the conjugate gradient method. Mathematical Programming 12 (1), 241–254.

Repoux, M., Boyaci, B., Geroliminis, N., 2014. An event-based simulation for optimising one-way carsharing systems. In: 14th Swiss Transport Research Conference, 14–16 May 2014, Monte Verità/Ascona, Switzerland.

Schmoller, S., Weikl, S., Muller, J., Bogenberger, K., 2014. Empirical data analysis of free-floating carsharing systems. In: Emerging and Innovative Public Transport and Technologies. CD-ROM. Transportation Research Board of the National Academies, Washington, D.C., USA.

Schon, C., 2008. Integrated airline schedule design, fleet assignment and strategic pricing. In: Multikonferenz Wirtscaftsinformatik (MKWI).

Schure, J., Napolitan, F., Hutchinson, R., 2012. Cumulative impacts of carsharing and unbundled parking on vehicle ownership & mode choice. Transportation Research Record: Journal of the Transportation Research Board 2319, 96–104.

Schuster, T., Byrne, J., Corbett, J., Schreuder, Y., 2005. Assessing the potential extent of carsharing – a new method and its applications. Transportation Research Record: Journal of the Transportation Research Board 1927, 174–181.

Shaheen, S., Cohen, A., Roberts, J., 2006. Carsharing in North America: market growth, current developments, and future potential. Transportation Research Record: Journal of the Transportation Research Board 1986, 116–124.

Shaheen, S., Sperling, D., Wagner, C., 1999. A short history of carsharing in the 90's. Journal of World Transport Policy and Practice 5 (3), 16–37.

Sinha, K., Labi, S., 2007. Transportation Decision Making: Principles of Project Evaluation and Programming. John Wiley & Sons, Chichester, United Kingdom.

Sioui, L., Morency, C., Trépanier, M., 2013. How carsharing affects the travel behavior of households: a case study of Montréal, Canada. International Journal of Sustainable Transportation 7 (1), 52–69.

Stützle, T., 2006. Iterated local search for the quadratic assignment problem. European Journal of Operational Research 174 (3), 1519–1539.

Stützle, T., Hoos, H. Analyzing the run-time behaviour of iterated local search for the TSP. In: Proceedings III Metaheuristics International Conference, 19–22 July 1999, Rio de Janeiro, Brazil, pp. 1–18.

Taplin, J., Hensher, D., Smith, B., 1999. Preserving the symmetry of estimated commuter travel elasticities. Transportation Research Part B 33 (3), 215–232.

Uesugi, K., Mukai, N., Watanabe, T., 2007. Optimization of vehicle assignment for car sharing system. Knowledge-Based Intelligent Information and Engineering Systems 4693, 1105–1111.

Wang, J., Lindsey, R., Yang, H., 2011. Nonlinear pricing on private roads with congestion and toll collection costs. Transportation Research Part B 45 (1), 9–40.

Weikl, S., Bogenberger, K., 2013. Relocation strategies and algorithms for free-floating carsharing systems. IEEE Intelligent Transportation Systems Magazine 5 (4), 100–111.

Wie, B., Tobin, R., 1998. Dynamic congestion pricing models for general traffic networks. Transportation Research Part B 32 (5), 313–327.

York, B., Fabricatore, D, 2001. Puget Sound Vanpool Marketing Assessment, Office of Urban Mobility, WSDOT.

Zhou, S., 2012. Dynamic Incentive Scheme for Rental Vehicle Fleet Management. MSc Thesis, Massachusetts Institute of Technology, Cambridge MA, USA.