Mathematical Model for the Study of Relocation Strategies in One-way Carsharing Systems

Aurélien Carlier, Alix Munier-Kordon, Witold Klaudel

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Transportation Research Procedia 00 (2015) 000–000

Transportation Research Procedia

www.elsevier.com/locate/procedia

18th Euro Working Group on Transportation, EWGT 2015, 14-16 July 2015, Delft, The Netherlands

Mathematical model for the study of relocation strategies in one-way carsharing systems

Aurélien Carlier [a,b,c], Alix Munier-Kordon [b], Witold Klaudel [a,c]

[a] *Renault SAS, Technocentre, 1 avenue du Golf, 78288 Guyancourt, France*
[b] *Sorbonne Universités, UPMC Univ Paris 6, UMR 7606, LIP6, F-75005, France*
[c] *IRT SystemX, 8 Avenue de la Vauve, 91120 Palaiseau, France*

Abstract

Carsharing is today considered as an ecological and innovative solution to improve urban mobility. The one-way version, where vehicles can be drop-off in any station, brings however some challenging open questions. The system has to be design as part of the global transportation one and vehicle relocation operations must be included to get the higher level of service.

In this paper, we consider a one-way carsharing system where stations and their location are fixed. The optimization problem consists in maximizing the total number of satisfied demands for a limited number of vehicles and relocation operations. We propose a formal definition and a mathematical model using Integer Linear Programming (ILP). We show that the problem size is strongly related to the number of possible relocation operations and a polynomial subcase is exhibited. Numerical results highlight that vehicle relocations can be drastically reduced without deteriorating the quality of solutions. Our method can thus be easily used in system management to evaluate possible implementation of vehicle relocation strategies.

Keywords: one-way carsharing, vehicle relocation strategies, transport optimization, mathematical modelling.

1. Introduction

In recent years, many efforts have been dedicated to understand how to organize our transportation systems. Due to many externalities such as pollution, congestion or excessive energy consumption, new alternatives have to be identified. Although public transports are a relevant option for mass transit, they are facing drawbacks with respect to public perception and flexibility. Moreover, the network design has been often done many decades ago when the urban setting and the demand distribution were different. We are now facing with a critical point where the network appears completely saturated and presents difficulties to absorb the demand.

E-mail address: aurelien.carlier@irt-systemx.fr, alix.munier@lip6.fr, witold.klaudel@renault.com

Since the early 1990s, carsharing systems emerged in many cities as an ecological and innovative solution to this dilemma. From a social and ecological point of view, carsharing reduces the average number of vehicles per household (Martin et al., 2010; Ter Schure et al., 2012) and the total number of vehicles on the road. Shared vehicles have higher utilization rates and are used more efficiently than private ones (Litman, 2000; Schuster et al., 2005). Some studies observe that carsharing improves traffic fluidity and produces many positive environmental effects, such as CO2 reductions (Martin and Shaheen, 2011).

The carsharing principle is to make available a fleet of vehicles distributed over a set of stations that can be used by a wide group of users (Shaheen et al., 1999). The return station, where commuters can (or have to) drop-off a vehicle, divides station-based systems into two categories. Round-trip systems require users to return vehicles to the station they were picked up, whereas one-way systems allow a distinct return station. System flexibility has been identified recently as the critical factor to join a carsharing system (Efthymiou et al., 2012), making this criterion a strong development catalyst. Consequently, free-floating carsharing systems have came up a few years ago with a new feature. There are no station and users can pick-up and drop-off the vehicles freely within a predefined area (Weikl and Bogenberger, 2012). Of course, for one-way and free-floating systems, the higher the flexibility is, the harder it is to manage the system.

The uneven nature of the trip pattern in urban areas leads to unbalanced situations that causes hard operational problems. To ensure a good system efficiency, vehicle relocation operations have emerged as a good solution to balance vehicle stock. As such, they have to be part of the system design, as well as the system management, bringing then new research challenges.

The intuitive approach for solving the vehicle unbalance problem is to consider that periodic relocation operations can be done by the operator. Kek et al. (2006) and Kek et al. (2009) have used discrete event simulation models to help operators manage their systems while minimizing available resources (such as vehicles and staff members) and ensuring a certain level of service. Fan et al. (2008) introduced a trip selection approach, deciding which vehicle reservations should be accepted or denied and how many vehicles should be relocated or held to maximise profit. More recently, Correia and Antunes (2012) developed a station location approach. The goal was to study the effect of stations' location on capturing more favorable trip pattern to balanced the vehicle distribution in the network. The idea was to transfer the system unbalance to the clients by decreasing their possibility of accessing this system. The authors concluded that the unbalance situation would lead to severe financial losses if all demand have to be satisfied, even if the trip is very expensive. They also found that financial losses could be reduced by making appropriate choices of the stations configuration (number, location and size), but profits could only be achieved with full control over trip selection.

In this study, we address an opened system design problem and an innovative approach to solve it. Considering potential one-way carsharing station locations and demands over time, what could be the optimal system configuration capturing the higher number of demands ? The paper is organized as follow. The next section introduces more precisely the optimization problem addressed in this work. Section 3 presents the problem modelling using time expanded graphs (TEGs). Section 4 formulates the problem as an Integer Linear Program (ILP). Section 5 is devoted to the study of a polynomial subcase. Section 6 is dedicated to experimentations and results on graph densities, solver computation times and optimal distances. Finally, section 7 finishes with some conclusions and directions for further research.

2. Problem definition

This section aims to present the optimization problem. Inputs, which consist mainly in the description of the travels and the demands, are first described briefly, followed by the objectives. The section ends with a formal definition of the optimal dimensioning problem.

2.1. Inputs

Let $\mathcal{N} = \{1, \cdots, N\}$ be the set of $N \in \mathbb{N}^\star$ stations. Since they are usually located in dense urban areas, their respective number of parking spaces are considered limited. Let $Z(i) \in \mathbb{N}^\star$ be the maximum station capacity of station $i \in \mathcal{N}$.

It is assumed that the system operates in a representative time period, such as a typical weekday or an average week. Let $\mathcal{H} = \{1, \cdots, T\}$ be this representative time period, discretized in $T \in \mathbb{N}^\star$ time steps. Periods are also supposed to be repeated, meaning that any time $t \in \mathbb{N}^\star$ is associated to $t' \in \mathcal{H}$ with $(t - t') \mod T = 0$.

All travel times are known between each couple of stations $(i, j) \in \mathcal{N}^2$ at every departure time taken in $\mathcal{H}$. We call in the next $\delta(i, j, t)$ the amount of time steps needed to reach station $j \in \mathcal{N}$ from station $i \in \mathcal{N}$ when departure time is $t \in \mathcal{H}$. It is assumed that, for any triple $(i, j, t) \in \mathcal{N}^2 \times \mathcal{H}$, $\delta(i, j, t) < T$.

Similarly, a set of M demands are fixed: each of them is designated by $D(i, j, t)$ where i and j are two stations picked in $\mathcal{N}$ and t is a time step belonging to $\mathcal{H}$.

2.2. Objectives

A feasible solution of our optimization problem is given by a set of vehicles, each of them associated with its position in the system at any-time during the period studied. The first objective is to maximize the total number of satisfied demands, *i.e.* for which a vehicle is allocated. However, two other objectives must be taken into account: each vehicle in the system is associated to a fixed cost, so that the total number of vehicles must be minimized. In the same way, vehicle relocations are fundamental for increasing the number of satisfied demands with a fixed number of vehicles. However, they cost an extra charge for the operator, and thus their number should also be limited. In the following, the total number of vehicles and relocations are referred respectively by C and R.

2.3. Formal problem statements

A formal definition of our main optimization problem, referred as the basic carsharing problem with relocations [BCPR] can be stated as follows:

Basic carsharing problem with relocations [BCPR]:

Inputs: A set of stations $\mathcal{N}$ with their capacity $Z(i)$, $i \in \mathcal{N}$, time periods set $\mathcal{H} = \{1, \cdots, T\}$, travel times $\delta(i, j, t)$ for each triplet $(i, j, t) \in \mathcal{N}^2 \times \mathcal{H}$, a set of M demands, fixed number of vehicles C and relocation operations R.

Question: What is the maximum number of demands $m \leq M$ that can be captured by a vehicle routing of at most C vehicles and R vehicle relocation operations during the considered period $\mathcal{H}$?

We will show that [BCPR] belongs to $\mathcal{NP}$ by modelling feasible solutions as a non classical flow problem.

3. Time Expanded Graph

Including time in a network flow model can be done using Time Expanded Graphs (TEGs) as suggested by Ahuja et al. (1993). Carsharing stations are duplicated at every discrete time-step so that links (arcs) between stations could represent time-dependent vehicle operations (staying parked, satisfying a demand or being relocated). Subsection 3.1 presents some basic notations, while subsection 3.2 describes precisely the partition of the arcs into 3 fixed sets. This section terminates with the calculation of the total time associated to a path or a circuit in the graph.

3.1. Basic definitions

The Time Expanded Graph is a valued directed graph $\mathcal{G} = (\mathcal{X}, \mathcal{A}, u)$ such that nodes represent stations states over the time period, *i.e.* $\mathcal{X} = \mathcal{N} \times \mathcal{H}$. Any arc $a = (x, y) \in \mathcal{A}$ is associated to a possible move of a vehicle from node x to y. The capacity $u(a)$ is the maximum number of vehicles allowed on a.

Let the function $\eta : \mathcal{X} \to \mathcal{N}$ with $x = (i, t) \mapsto \eta(x) = i$ referring to the station of a node $x \in \mathcal{X}$. Similarly, $\theta : \mathcal{X} \to \mathcal{H}$ with $x = (i, t) \mapsto \theta(x) = t$ is the step-time of x. Let $\Gamma^-(\mathcal{G}, x)$ and $\Gamma^+(\mathcal{G}, x)$ respectively denotes the set of immediate predecessors and successors of a node $x \in \mathcal{X}$ in $\mathcal{G}$, *i.e.* $\Gamma^-(\mathcal{G}, x) = \{y \in \mathcal{X} \mid (y, x) \in \mathcal{A}\}$ and $\Gamma^+(\mathcal{G}, x) = \{y \in \mathcal{X} \mid (x, y) \in \mathcal{A}\}$. We simply note $\Gamma^-(\mathcal{G}, x) = \Gamma^-(x)$ and $\Gamma^+(\mathcal{G}, x) = \Gamma^+(x)$ if no confusion is possible.

3.2. Set of arcs

Arcs set $\mathcal{A}$ is partitioned into three sets $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{A}_3$ defined as follows.

- $\mathcal{A}_1$ is a set of arcs associated to vehicles staying in a same station between two consecutive time steps. Formally,

$$\mathcal{A}_1 = \{(x, y) \in \mathcal{X}^2 \mid \eta(x) = \eta(y) \textit{ and } \theta(y) = \theta(x) + 1 \bmod T\}. \tag{1}$$

  The capacity of any arc $a = (x, y) \in \mathcal{A}_1$ with $i = \eta(x) = \eta(y) \in \mathcal{N}$ is $u(a) = Z(i)$.
- Any arc $a = (x, y) \in \mathcal{A}_2$ corresponds to a positive demand from $\eta(x)$ to $\eta(y)$ at time $\theta(x)$. The arrival time is $\theta(x) + \delta(\eta(x), \eta(y), \theta(x)) \mod T$. Arcs set $\mathcal{A}_2$ is then formally defined as

$$\mathcal{A}_2 = \{(x, y) \in \mathcal{X}^2 \mid D(\eta(x), \eta(y), \theta(x)) > 0 \textit{ and } \theta(x) + \delta(\eta(x), \eta(y), \theta(x)) = \theta(y) \bmod T\}. \tag{2}$$

  The capacity of any arc $a = (x, y) \in \mathcal{A}_2$ equals $u(a) = D(\eta(x), \eta(y), \theta(x))$.
- Elements from $\mathcal{A}_3$ model relocations. Each arc $a = (x, y) \in \mathcal{A}_3$ is associated to a possible relocation from station $\eta(x)$ to $\eta(y)$ at time $\theta(x)$:

$$\mathcal{A}_3 = \{(x, y) \in \mathcal{X}^2 \mid \eta(x) \neq \eta(y) \textit{ and } \theta(x) + \delta(\eta(x), \eta(y), \theta(x)) = \theta(y) \bmod T\}. \tag{3}$$

  The capacity $u(a)$ of any arc $a = (x, y) \in \mathcal{A}_3$ is not bounded.

The total number of arcs is then $|\mathcal{A}| = \sum_{k=1}^{3} |\mathcal{A}_k| = N \cdot T + M + (N \cdot T) \cdot (N - 1) = N^2 \cdot T + M$. As $M \ll N^2$, $|\mathcal{A}| = \Theta(N^2 \cdot T)$. We observe that $|\mathcal{A}_3| \gg |\mathcal{A}_1 \cup \mathcal{A}_2|$ and that the number of arcs is proportional to $\mathcal{A}_3$.

3.3. Effective duration of a path or a circuit

The duration of any path of a TEG may be easily evaluated. Indeed, for any couple of time instants $(t, t') \in \mathcal{H}^2$, let the function $\vartheta : \mathcal{H}^2 \mapsto \mathbb{N}^\star$ that computes the number of time-steps between those two instants. It is defined formally as

$$\vartheta(t, t') = \begin{cases} t' - t & \text{if } t \leq t' \\ T + t' - t & \text{otherwise.} \end{cases}$$

For each arc $a = (x, y) \in \mathcal{A}$, let us define and set the boolean value ϵ_a to true if $\theta(x) > \theta(y)$. For any arc $a = (x, y) \in \mathcal{A}$, the effective time required for a move from x to y is then equal to $\ell(a) = \theta(y) - \theta(x) + \epsilon_a \cdot T$. By extension, if $\mu = (a_1, \cdots, a_p) \in \mathcal{A}^p$ is a path of the TEG from x to y, the value $\ell(\mu) = \sum_{i=1}^{p} \ell(a_i)$ is the total time required for a vehicle going from x to y following μ. Next lemma evaluates the total time of any circuit c.

Lemma 1. *The total time of any circuit $c = (a_1, \cdots, a_p)$ is $\ell(c) = T \times \sum_{i=1}^{p} \epsilon_{a_i}$.*

Proof. Let x_i, $i \in \{1, \cdots, p + 1\}$ be the sequence of elements from $\mathcal{X}$ such that, $x_{p+1} = x_1$ and $\forall i \in \{1, \cdots, p\}$, $a_i = (x_i, x_{i+1})$. The total time of c is then

$$\ell(c) = \sum_{i=1}^{p} \ell(a_i) = \sum_{i=1}^{p} (\theta(x_{i+1}) - \theta(x_i) + \epsilon_{a_i} \cdot T) = T \times \sum_{i=1}^{p} \epsilon_{a_i},$$

the result. ■

4. Modelling of the carsharing problem with relocations

This section is devoted to the modelling of our main optimization problem [BCPR] using an Integer Linear Program (ILP in short). Subsection 4.1 shows that any feasible solution may be equivalently expressed using flow variables on the arcs of the current TEG. It is proved in Subsection 4.2 that our objective can be expressed as linear functions of flow variables, concluding to the modelling of our optimization problem by an ILP.

4.1. Decision variables

The aim of our study is to compute the planning of each vehicles during the period. At any time, each of them is either parked in a station or in transit between two stations. Its position over the period can be modelled as a vehicle tour *i.e.* a circuit $c = (a_1, \cdots, a_p)$ in the TEG.

We show in the following that a feasible solution can be described by only considering the number of vehicles passing through each arc. For each arc $a = (x, y) \in \mathcal{A}$, we call $\varphi(a)$ the flow of vehicles transiting through the arc a. It can be interpreted as the number of vehicle staying in station $\eta(x)$ between two consecutive time-steps $\theta(x)$ and $\theta(y)$ if $a \in \mathcal{A}_1$, or the number of vehicle moving from station $\eta(x)$ at time $\theta(x)$ to station $\eta(y)$ otherwise.

Since the total number of vehicles transiting to any node $x \in \mathcal{X}$ is constant,

$$\sum_{y\in\Gamma^-(x)} \varphi((y, x)) = \sum_{y\in\Gamma^+(x)} \varphi((x, y)). \tag{4}$$

A flow $\varphi : \mathcal{A} \mapsto \mathbb{N}$ is said to be feasible if $\forall a \in \mathcal{A}$, $\varphi(a) \leq u(a)$ and $\forall x \in \mathcal{X}$, the flow conservation equation (4) is true. A feasible flow may be easily obtained from any feasible set of vehicle tours.

We prove in the following that the reverse is also true, with the consequence that any feasible solution of our problem can be described using a flow. Next lemma computes the exact number of vehicles associated to a constant unitary flow over a circuit c.

Lemma 2. *Let c be a circuit and φ_c a feasible flow such that:*

$$\varphi_c(a) = \begin{cases} 1 & \text{if } a \text{ belongs to } c \\ 0 & \text{otherwise.} \end{cases}$$

The minimum number of vehicles to insure φ_c is $\frac{\ell(c)}{T}$.

Proof. For any time value $t \in \mathcal{H}$, let us define the set $C_t(c)$ as the arcs $a = (x, y)$ from c starting at time t or earlier but ending after t. Since $\vartheta(\theta(x), t)$ equals the number of time steps from $\theta(x)$ to t, we get $C_t(c) = \{a = (x, y) \in c \mid \vartheta(\theta(x), t) < \ell(a)\}$.

Now, since c is a circuit, the value $|C_t(c)|$ is a constant $\forall t \in \mathcal{H}$ and corresponds to the total number of vehicles needed to insure a unitary flow over c. Let us prove that $|C_T(c)| = \sum_{a\in c} \epsilon_a$. For that purpose, setting $B(c) = \{a = (x, y) \in c \mid \epsilon_a = 1\}$, we show that $B(c) = C_T(c)$.

- $B(c) \subseteq C_T(c)$: if $a = (x, y) \in B(c)$, then as $\theta(x) \leq T$, $\vartheta(\theta(x), T) = T - \theta(x)$. Now, since $\epsilon_a = 1$ and $\theta(y) \geq 1$, $\ell(a) = \theta(y) - \theta(x) + T \geq 1 - \theta(x) + T > \vartheta(\theta(x), T)$ and $a \in C_T(c)$.
- $C_T(c) \subseteq B(c)$: let consider now an arc $a = (x, y) \in C_T(c)$. Since $\vartheta(\theta(x), T) = T - \theta(x) < \ell(a)$ we get that $\theta(y) - \theta(x) + \epsilon_a \cdot T > T - \theta(x)$ and thus $\theta(y) + \epsilon_a \cdot T > T$. As $\theta(y) \leq T$, we necessarily have $\epsilon_a = 1$ and thus $a \in B(c)$.

Now, by Lemma 1, $|C_T(c)| = \sum_{a\in c} \epsilon_a = \frac{\ell(c)}{T}$, the lemma. ■

Theorem 1. *Any feasible solution φ can be decomposed into a set of circuits $\mathcal{S}$ such that, for any arc $a \in \mathcal{A}$, $\varphi(a) = \sum_{c\in\mathcal{S}} \varphi_c(a)$.*

Proof. The proof is by recurrence on $n(\varphi) = \sum_{a\in\mathcal{A}} \varphi(a)$. The theorem is trivially true if $n(\varphi) = 0$.

Let suppose now that $n(\varphi) > 0$, thus there exists at least one arc $a = (x, y) \in \mathcal{A}$ with $\varphi(a) > 0$. Set $\mu_0 = (x, y)$ and let consider the sequence of paths μ_i built as follows:

1. Stop the sequence as soon as μ_i contains a circuit c;
2. Otherwise, let $\tilde{a} = (\tilde{x}, \tilde{y})$ the last arc of μ_i. Since $\varphi(\tilde{a}) > 0$, the flow conservation equation (4) insures that there exists an arc a starting at $\tilde{y}$ with $\varphi(a) > 0$. We then set $\mu_{i+1} = \mu_i \cdot a$.

As $\mathcal{G}$ has a finite number of nodes, the algorithm stops and a non empty circuit c is returned. The flow $\hat{\varphi}$ defined as

$$\hat{\varphi}(a) = \begin{cases} \varphi(a) - 1 & \text{if } a \in c \\ \varphi(a) & \text{otherwise.} \end{cases}$$

is feasible with $n(\hat{\varphi}) < n(\varphi)$, thus the theorem. ■

Note that the number of flow variables is a polynomial function on the size of the problem. This is not true anymore for vehicle tours, which number can be exponential. The consequence is that the determination of a flow is in $\mathcal{NP}$, which is not the case for the determination of vehicle tours.

4.2. Mathematical model

The aim of this part is to model our optimization problem using flow variables. The total number of fulfilled demands and relocations are linearly computable for any fixed flow. Next theorem shows that the total number of vehicles is also a linear function of the flow:

Theorem 2. *The minimum total number of cars required for a feasible flow φ equals $\sum_{a \in \mathcal{A}} \varphi(a) \cdot \epsilon_a$.*

Proof. Let $\mathcal{S}$ be a set of circuits obtained from the decomposition of φ following Theorem 1 and let V be the minimum number of cars associated with φ. By Lemmas 1 and 2, the total number of cars of any circuit $c \in \mathcal{S}$ is

$$\frac{\ell(c)}{T} = \sum_{a \in c} \epsilon_a = \sum_{a \in \mathcal{A}} \epsilon_a \cdot \varphi_c(a).$$

Since vehicles are allocated to exactly one circuit, the total number V of vehicles is then

$$V = \sum_{c \in \mathcal{S}} \sum_{a \in \mathcal{A}} \epsilon_a \cdot \varphi_c(a).$$

Now, from Theorem 1, $\varphi(a) = \sum_{c \in \mathcal{S}} \varphi_c(a)$. Thus,

$$V = \sum_{c \in \mathcal{S}} \sum_{a \in \mathcal{A}} \epsilon_a \cdot \varphi_c(a) = \sum_{a \in \mathcal{A}} \epsilon_a \cdot \sum_{c \in \mathcal{S}} \varphi_c(a) = \sum_{a \in \mathcal{A}} \varphi(a) \cdot \epsilon_a,$$

the theorem. ■

The modeling of our main optimization problem [BCPR] follows. Remind that R and C are fixed bounds for respectively the total number of relocation operations and vehicles. Equation (5) is the maximization of the total demand. Equation (6)and (7) express respectively the bound on the total number of relocations and the total number of vehicles. Equations (8), (9) and (10) are lastly flow constraints.

$$\max \quad \sum_{a \in \mathcal{A}_2} \varphi(a) \tag{5}$$

$$s.t. \begin{cases} \sum_{a \in \mathcal{A}_3} \varphi(a) \leq R & (6) \\ \sum_{a \in \mathcal{A}} \varphi(a) \cdot \epsilon_a \leq C & (7) \\ \varphi(a) \leq u(a) \qquad \forall a \in \mathcal{A} & (8) \\ \sum_{y \in \Gamma^-(x)} \varphi((y, x)) = \sum_{y \in \Gamma^+(x)} \varphi((x, y)) \qquad \forall x \in \mathcal{X} & (9) \\ \varphi(a) \in \mathbb{N} \qquad \forall a \in \mathcal{A} & (10) \end{cases}$$

The total number of equations is around $2|\mathcal{A}| + |\mathcal{X}| = N^2 \cdot T + M + N \cdot T = \Theta(N^2 \cdot T)$. The size of the ILP is thus proportional to the number of relocation arcs.

5. A polynomial subcase

This section aims to prove that the determination of a flow that satisfies all the demands without a constraint on the total number of relocations or vehicles is a polynomial problem. The formal definition of this problem, designated by ALL-DEMANDS, follows.

[ALL-DEMANDS]:

Inputs : A Time Expanded Graph $\mathcal{G} = (\mathcal{X}, \mathcal{A}, u)$.

Question : Is there a feasible flow φ such that all the demands are fufilled, *i.e* $\forall a \in \mathcal{A}_2, \varphi(a) = u(a)$?

Let I be an instance of ALL-DEMANDS. We associate an instance of a max-flow problem $f(I)$ which network $\widehat{\mathcal{G}} = (\widehat{\mathcal{X}}, \widehat{\mathcal{A}}, \widehat{w})$ is defined as follows:

1. Vertices are $\widehat{\mathcal{X}} = \mathcal{X} \cup \{s^\star, t^\star\} \cup \{s_a, t_a, a \in \mathcal{A}_2\}$. $s^\star$ and $t^\star$ are respectively the source and the sink of $\widehat{\mathcal{G}}$, while s_a and t_a are two additional vertices associated to any demand arc $a \in \mathcal{A}_2$.
2. Arcs set is $\widehat{\mathcal{A}} = \mathcal{A}_1 \cup \mathcal{A}_3 \cup \{(x, t_a), (t_a, t^\star), (s^\star, s_a), (s_a, y), \forall a = (x, y) \in \mathcal{A}_2\}$.
3. Maximum capacity of arcs are $\widehat{w}(a) = u(a)$ for $a \in \mathcal{A}_1 \cup \mathcal{A}_3$. Otherwise, for any arc $a = (x, y) \in \mathcal{A}_2$, $\widehat{w}((x, t_a)) = \widehat{w}((t_a, t^\star)) = \widehat{w}((s^\star, s_a)) = \widehat{w}((s_a, y)) = u(a)$.

Note that this transformation is a polynomial function and does not depend on the structure of $\mathcal{G}$.

Theorem 3. *Let an instance of* ALL-DEMANDS *expressed by a TEG $\mathcal{G}$. There exists a feasible flow fulfilling all the demands of $\mathcal{G}$ if and only if there exists a maximum flow in $\widehat{\mathcal{G}}$ of value $\sum_{a\in\mathcal{A}_2} u(a)$.*

Proof. Let suppose that φ is a feasible flow of $\mathcal{G}$ that fulfils all the demands, *i.e.* for any arc $a \in \mathcal{A}_2$, $\varphi(a) = u(a)$. A flow $\widehat{\varphi}$ of $\widehat{\mathcal{G}}$ may be built as follows:

1. $\forall a \in \mathcal{A}_1 \cup \mathcal{A}_3, \widehat{\varphi}(a) = \varphi(a)$;
2. For any arc $a = (x, y) \in \mathcal{A}_2$, $\widehat{\varphi}((x, t_a)) = \widehat{\varphi}((t_a, t^\star)) = \widehat{\varphi}((s^\star, s_a)) = \widehat{\varphi}((s_a, y)) = u(a)$.

We prove that $\widehat{\varphi}$ is a feasible flow of $\widehat{\mathcal{G}}$ of value $\sum_{a\in\mathcal{A}_2} u(a)$. Indeed, let consider a node $x \in \widehat{\mathcal{X}}$.

1. Let suppose first that $x \in \mathcal{X}$. Then any demand arc $a = (y, x) \in \mathcal{A}_2$ (*resp.* $a = (x, y) \in \mathcal{A}_2$) of flow $\varphi(a)$ is associated in $\widehat{A}$ to an arc $e = (s_a, x)$ (*resp.* $e = (x, t_a)$) with $\widehat{\varphi}(e) = \varphi(a)$. Thus,
$$\sum_{a\in\Gamma^-(\widehat{\mathcal{G}},x)} \widehat{\varphi}(a) = \sum_{a\in\Gamma^-(\mathcal{G},x)} \varphi(a) = \sum_{a\in\Gamma^+(\mathcal{G},x)} \varphi(a) = \sum_{a\in\Gamma^+(\widehat{\mathcal{G}},x)} \widehat{\varphi}(a).$$
2. For any arc $a = (z, y) \in \mathcal{A}_2$, the two vertices t_a and s_a are such that
$$\sum_{e\in\Gamma^-(\widehat{\mathcal{G}},t_a)} \widehat{\varphi}(e) = \widehat{\varphi}((x, t_a)) = \widehat{\varphi}((t_a, t^\star)) = \sum_{e\in\Gamma^+(\widehat{\mathcal{G}},t_a)} \widehat{\varphi}(e) \text{ and } \sum_{e\in\Gamma^-(\widehat{\mathcal{G}},s_a)} \widehat{\varphi}(e) = \widehat{\varphi}((s^\star, s_a)) = \widehat{\varphi}((s_a, y)) = \sum_{e\in\Gamma^+(\widehat{\mathcal{G}},s_a)} \widehat{\varphi}(e).$$
3. Lastly,
$$\sum_{e\in\Gamma^-(\widehat{\mathcal{G}},t^\star)} \widehat{\varphi}(e) = \sum_{a\in\mathcal{A}_2} u(a) \text{ and } \sum_{e\in\Gamma^+(\widehat{\mathcal{G}},s^\star)} \widehat{\varphi}(e) = \sum_{a\in\mathcal{A}_2} u(a).$$

The consequence is that $\widehat{\varphi}$ is a feasible flow of $\widehat{\mathcal{G}}$ of value $\sum_{a\in\mathcal{A}_2} u(a)$.

Conversely, any feasible flow of $\widehat{\mathcal{G}}$ of value $\sum_{a\in\mathcal{A}_2} u(a)$ verifies that, for any arc $a = (x, y) \in \mathcal{A}_2$, $\widehat{\varphi}((x, t_a)) = \widehat{\varphi}((t_a, t^\star)) = \widehat{\varphi}((s^\star, s_a)) = \widehat{\varphi}((s_a, y)) = u(a)$. A feasible flow for $\mathcal{G}$ can be easily obtained by setting:

1. $\forall a \in \mathcal{A}_1 \cup \mathcal{A}_3, \varphi(a) = \widehat{\varphi}(a)$;

2. For any arc $a = (x, y) \in \mathcal{A}_2$, $\varphi(a) = u(a)$,

the theorem. ■

According to Ahuja et al. (1993), the existence of a maximum-flow of a fixed value is a polynomial problem. The following corollary is thus a consequence of Theorem 3:

Corollary 1. ALL-DEMANDS *is polynomial.*

6. Experimentations

Previous results presented in Carlier et al. (2014) aimed to evaluate the scalability of the optimization model described before. Computation times were obtained on small realistic instances with 10 stations and 144 time steps using an open source solver (GLPK). The authors observed that they were negligible compared to model building time and decided to study the building time behaviour when the problem grows. Results shown that they were unaffordable in an industrial or real case context, even for a realistic case, and can be improved by reducing the size of the linear programs. Authors suggested to use better relocation strategies in order to reduce the graph density and problem size.

The aim of this section is to study the computation times for solving exactly or approximately the optimization problem for real case instances. More precisely, following previous experiments, we show that the density of relocation arcs drastically decreases computation times without impacting the quality of the solution.

The experimental conditions are first presented, followed by the description of the strategies for reducing the graph density. The section ends with the impacts on solver computation time and optimal distances to the baseline situation.

6.1. Experimental conditions

The generator described in Carlier et al. (2014) was considered to produce randomly realistic urban data taking into account demand variability over time and travel time penalties during pick hours. The size of instances is fixed to 50 stations, which corresponds to a reasonnable size for a real-life problem. 18 stations are arbitrarily placed in a dense and small “downtown” area whereas 32 are positioned in a peri-urban area. A total of 500 carsharing demands (requests) are randomly generated over a typical 24 hours week day period segmented in 144 time-steps of 10 minutes. To be realistic, 80% (*resp.* 60%) of the generated demand during rush hours is oriented from the suburbs (*resp.* the center) to the center (*resp.* the suburbs). Morning rush is set between 7 and 9 while evening rush is between 17 and 20. Travel times between stations are also given using an average car speed of 70km/h, applying a 160% penalty coefficient if the trip is done during the rush period.

All the computational results presented in this paper, including the random data generation, were made using an Intel(R) Core(TM) i5-3337U CPU @1.80GHz. Mathematical programs were solved using the Java API of IBM ILOG CPLEX 12.5.1.

6.2. Stategies for reducing graph density

Baseline situation (BS is short) corresponds to instances for which relocations are generated at each times steps, *i.e.* every 10 minutes. As pointed before, our idea is to reduce the number of relocation arcs, which corresponds in this case to more than 97% of the total number of arcs. The goal is to accelerate the resolution of our optimization problem.

Table 1 presents some numerical parameters of any generated TEG. We studied four strategies (S1 to S4) which generate relocation operations respectively every 30 minutes, 1, 2 and 4 hours. The number of nodes is always equal to $\mathcal{X} = N \times T = 50 \times 144 = 7200$, while the total number of arcs $|\mathcal{A}|$ decreases following the frequency of relocation operations (RF) during a day. The respective ratios of remaining arcs compared to BS (% BS) and the arcs removed (% elim) are also presented, followed by the exact number of relocation arcs (RA) and their proportion in the graph.

Reducing the number of relocation arcs decreases drastically the graph density: from - 65% for (S1) to almost - 94% for (S4). The resulting number of arcs for the strategy (S4) represents for instance $6,21\%$ of the BS one. Also observe that relocation arcs remain predominant, even for strategy (S4) for which they represent above 65% of the total number of arcs.

Table 1. Graph density depending on relocation strategies

Strategy	$\|\mathcal{N}\|$	$\|\mathcal{A}\|$	RF	% BS	% elim	RA	RA/$\|\mathcal{A}\|$
(BS) : Every 10 min	7200	360500	144	100, 00%	–	352800	97, 86%
(S1) : Every 30 min	7200	125300	48	34, 76%	- 65, 24%	117600	93, 85%
(S2) : Every 1h	7200	66500	24	18, 45%	- 81, 55%	58800	88, 42%
(S3) : Every 2h	7200	37099	12	10, 29%	- 89, 71%	29400	79, 25%
(S4) : Every 4h	7200	22400	6	6, 21%	- 93, 79%	14700	65, 63%

6.3. Improving solving times and evaluating optimal distances to baseline situation

The following experiments aim to evaluate the impact of the relocation reduction on computation time and solution quality of our optimization problem. A total of 30 instances were randomly generated. The five strategies were evaluated for each of them by solving 81 linear programs corresponding to the combinational values of R and C, set in this study in the range $\{0, 10, \cdots, 80\}$.

Table 2 summarizes our results solving approximately the integer linear program (LP in short) by considering real variables, while Table 3 shows the results using exact method (ILP in short). In both case, columns "μ_t", "min" and "max" reports respectively the average, the minimum and the maximum computation times in milliseconds while "σ_t" is the standard deviation. Columns "μ_d" and "σ_d" indicate the average distance to the solution obtained for (BS) and the corresponding standard deviation.

Table 2. Solver computation times and distances to the baseline situation for the LP version

	Solver computation times (mili sec)					Optimal distance to BS	
Strategy	μ_t	gain	σ_t	min	max	μ_d	σ_d
(BS) : Every 10 min	7669	–	6278	262	24406	–	–
(S1) : Every 30 min	2431	- 68%	1736	120	7347	0, 40%	0, 61%
(S2) : Every 1h	1325	- 82%	965	63	3555	1, 07%	1, 53%
(S3) : Every 2h	540	- 92%	358	36	1742	2, 56%	3, 65%
(S4) : Every 4h	323	- 95%	238	41	1608	5, 15%	6, 42%

Table 3. Solver computation times and distances to the baseline situation for the ILP version

	Solver computation times (mili sec)					Optimal distance to BS	
Strategy	μ_t	gain	σ_t	min	max	μ_d	σ_d
(BS) : Every 10 min	57949	–	35509	256	138123	–	–
(S1) : Every 30 min	9472	- 83%	6355	126	29963	0, 38%	0, 59%
(S2) : Every 1h	3059	- 94%	2131	69	10733	1, 01%	1, 45%
(S3) : Every 2h	918	- 98%	641	48	3685	2, 60%	3, 61%
(S4) : Every 4h	426	- 99%	251	30	1528	4, 92%	6, 18%

First note that computation times remain reasonable, even for the exact method with baseline strategy (always less than 139 seconds). However, the density reduction allows to decrease dramatically computation times. This reduction is particularly important for the exact method ILP. Also note that computation time values are almost equal for both exact or approximate solutions using (S4) strategy.

The surprise is that the optimal number of fulfilled demands are almost not impacted by the reduction of relocation arcs. The optimal values following the different strategies remain very close to the baseline one for both LP or ILP resolutions, with a gap varying from 0, 4% in (S1) to 5% in (S4).

7. Conclusion

This paper presents an original mathematical model for carsharing system design purposes. Our model is based on flows in a time expanded graph in order to group the vehicles passing through any road at any time. It is shown that vehicle routes can be recovered from any feasible flows. The main theoretical consequence is that our optimization problem belongs to $\mathcal{NP}$. An original polynomial sub-case, where all demands must be fulfilled has been also presented.

We shown experimentally that any random generated problem with realistic size (50 stations and 144 time steps) can be exactly solved within a reasonable time. We can withdraw that adopting a relocation strategy based on fixed time steps allow to handle larger problems, reduce graph density and largely improve solver calculation times while keeping good quality solutions.

The main conclusion is that our model can be consider to study vehicle relocation strategies in a real life context. We also experimentally proved that decreasing the relocation frequency has a few impact on the total number of satisfied demands. Scheduling operations every two hours for instance could help the operator organizing vehicle relocation routes while keeping a good level of service. The next step will consist in finding more advanced and flexible relocation strategies and including other operational constraints such as a limited number of employees in charge of the vehicle relocation task.

Acknowledgements

This research work has been carried out in the framework of the Technological Research Institute SystemX, and therefore granted with public funds within the scope of the French Program “Investissements d’Avenir”.

References

Ahuja, R. K., Magnanti, T. L., Orlin, J. B., 1993. Network flows: theory, algorithms, and applications. Prentice Hall, Englewood Cliffs, N.J.

Carlier, A., Munier-Kordon, A., Klaudel, W., nov 2014. Optimization of a one-way carsharing system with relocation operations. In: 10th International Conference on Modeling, Optimization and Simulation, MOSIM.

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

Efthymiou, D., Antoniou, C., Waddell, P., 2012. Which factors affect the willingness to join vehicle sharing systems? evidence from young greek drivers. In: Transportation Research Board 91st Annual Meeting. No. 12-1076.

Fan, W. D., Machemehl, R. B., Lownes, N. E., 2008. Carsharing: Dynamic decision-making problem for vehicle allocation. Transportation Research Record: Journal of the Transportation Research Board 2063 (1), 97104.

Kek, A. G., Cheu, R. L., Chor, M. L., 2006. Relocation simulation model for multiple-station shared-use vehicle systems. Transportation Research Record: Journal of the Transportation Research Board 1986 (1), 81–88.

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

Litman, T., 2000. Evaluating carsharing benefits. Transportation Research Record: Journal of the Transportation Research Board 1702 (1), 31–35.

Martin, E., Shaheen, S. A., Lidicker, J., 2010. Impact of carsharing on household vehicle holdings. Transportation Research Record: Journal of the Transportation Research Board 2143 (1), 150–158.

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

Schuster, T. D., Byrne, J., Corbett, J., Schreuder, Y., 2005. Assessing the potential extent of carsharing: A new method and its implications. Transportation Research Record: Journal of the Transportation Research Board 1927 (1), 174–181.

Shaheen, S. A., Sperling, D., Wagner, C., 1999. A short history of carsharing in the 90’s. Institute of Transportation Studies.

Ter Schure, J., Napolitan, F., Hutchinson, R., 2012. Cumulative impacts of carsharing and unbundled parking on vehicle ownership and mode choice. Transportation Research Record: Journal of the Transportation Research Board 2319 (1), 96–104.

Weikl, S., Bogenberger, K., 2012. Relocation strategies and algorithms for free-floating car sharing systems. In: 15th International IEEE Conference on Intelligent Transportation Systems.