FORMULATIONS FOR OPTIMAL SHARED OWNERSHIP AND USE OF AUTONOMOUS OR DRIVERLESS VEHICLES

Neda Masoud
Ph.D. Candidate
Department of Civil and Environmental Engineering and
Institute of Transportation Studies
University of California, Irvine
Irvine, CA, USA, 92697
Email: nmasoud@uci.edu
Corresponding author

R. Jayakrishnan
Professor
Department of Civil and Environmental Engineering and
Institute of Transportation Studies
University of California, Irvine
Irvine, CA, USA, 92697
Email: rjayakri@uci.edu

ABSTRACT
Advances in the automobile industry and intelligent transportation systems in the recent decade have made what was once a dream, driver-less vehicles, closer than ever to reality now. Although autonomous vehicles introduce many benefits ranging from decreasing delay to higher levels of safety on roads, they will be priced relatively high once they enter the market. The high price and the consequent low demand may translate to less motivation for the automobile industry to move toward mass production, and it could take decades for the market to reach equilibrium. In this paper, we describe formulations for analyzing shared ownership and use of autonomous vehicles as well as some variants that we newly propose. Households interested in participating in the program will join together, forming clusters of households. Each cluster will share the ownership of a set of autonomous vehicles. The program also allows participants to rideshare together. Such a program will decrease the number of vehicles needed by households, and will therefore make the ownership of autonomous vehicles more economical. In addition, clusters of households can register their vehicles in a carsharing program when they are not being used, in order to partially cover the ownership cost. We implement this program for a sample of households in San Diego, California, and discuss the reduction in vehicle ownership as a result of participating in the program.

Keywords: Autonomous, Driverless, Self-driving, Shared Ownership, Sharing Economy, Fractional Ownership, Shared Use, Carsharing, Ridesharing

INTRODUCTION

Sharing economy, also known as collaborative consumption, is a fairly old concept that focuses on the benefits obtained from sharing resources (products or services) that would otherwise go unused. Although communities have been using the concept of sharing economy locally for many years, advent of internet has led to its spread in global populations, and highlighted its benefits.

The sharing economy model has been historically used for high-value commodities, such as exotic automobiles, yachts, private jets, vacations homes, and the like. Curvy Road, for example, an exotic carshare company founded in 2000, provides fractional ownership of high-end vehicles in four cities in the US *(1)*. Although it has been long realized that taking ownership of under-utilized high-value assets may not be always economically wise, this economic model has become more popular recently for less expensive resources as well, thanks to new platforms that allow easy and quick development of companion mobile applications

Autonomous (or driver-less) vehicles are expected to enter the market in the near future. Although these vehicles introduce many benefits such as higher degree of safety and lower delays to the users and the transportation system in general, their high prices can be prohibitive when it comes to purchasing them. On the other hand, autonomous vehicles can decrease the total number of vehicles needed to perform daily tasks, since these vehicles can drive themselves to locations where there is demand for transportation. One possible strategy to make autonomous vehicles more affordable is to encourage shared ownership of these vehicles.

In addition to shared ownership, it is possible to decrease the ownership cost of autonomous vehicles further by (1) shared use of these vehicles, and (2) renting out the vehicles when they are not being used by owners.

Since autonomous vehicles can drive themselves, owning these vehicles decreases the number of vehicles a household requires to perform daily tasks. Figure 1 shows the average daily vehicle miles travelled (VMT) by each vehicle in a household in the US in 2009. This figure suggests that the higher the number of vehicles owned by a household, the less the average use of each additional vehicle would be. Although for a typical household owning more than one vehicle might be financially justifiable considering the level of comfort and peace of mind it might bring, this justifiability decreases with the purchase of additional vehicles. Apart from the initial investment (or monthly payments), the cost of insurance, depreciation of value, and parking can turn vehicle ownership into a financial burden. With autonomous vehicles, fewer vehicles can cover the same trips compared to a higher number of regular vehicles.

In this paper, we introduce analytical optimization schemes to study shared vehicle ownership and use program, in which a group of households jointly own and use a set of autonomous vehicles. These households can share rides with each other if the spatiotemporal distribution of their trips allow that. Though no rigorous analysis under optimal operations appear to have been done in the literature yet, there is some awareness of such possibilities in the automotive industry as well as among the increasing number of researchers and aficionados of autonomous vehicles *(2, 3, 4, 5)*. In this paper, along with an analytical formulation, we also offer certain new possibilities such as renting the vehicles out when no one is using them.

The analysis presented here are equally applicable to both fully autonomous vehicles and driverless vehicles (which may not be technically autonomous), the distinction between the two terms being relatively well-known now. In the following discussions, however, we use only the term "autonomous" to avoid unnecessary repetition of the fact that it refers to driverless vehicles as well.

LITERATURE REVEIEW

The proposed model in this paper combines three shared mobility alternatives: fractional vehicle ownership, and peer-to-peer car- and ride-sharing. We combine the individual advantages offered by each of these models, and propose a system that attempts to maximize efficiency.

Fractional ownership of luxury commodities emerged in the US in 1970s with real estate time shares, and was later spread to other high value commodities *(6)*. Ford, in partnership with Zoomcar, is the first company to start a pilot project of fractional ownership of non-luxury vehicles in Bengaluru, India, as a part of its 25 mobility experiment initiative *(7)*. During this three month pilot which is planned to take off in 2015, Zoomcar will provide 5 vehicles, each of which will be shared by 6 individuals. This is a first step for Ford and Zoomcar to study the impacts and implications of fractional vehicle ownership, and Ford is planning to build upon this first step in the future *(8)*.

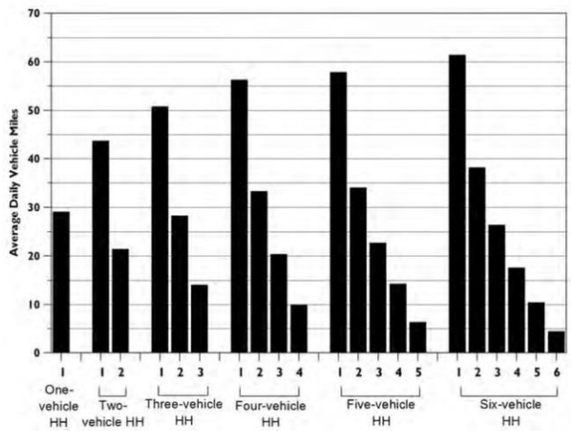


FIGURE 1 Average daily vehicle miles for households with various number of vehicles. The figure is from the National Household Travel Survey (NHTS), 2009.

In the proposed system in this paper, households who share the ownership of vehicles have the possibility of sharing rides, if their trips are spatiotemporally compatible. Ridesharing systems are well-studied in the literature, and a large volume of studies have confirmed their numerous advantages, including savings in vehicle miles travelled, and less damage to the environment *(9, 10)*.

In order to make shared ownership of autonomous vehicles more affordable, households can rent their vehicles when they are not being used. The advantage of using autonomous vehicles in carsharing programs is that the complicated dispatching problem that one-way carsharing systems face does not arise. A good review and classification of carsharing models can be found in Barth, M., and S. A. Shaheen *(11)*. There are multiple studies that link carsharing to reduction in household vehicle holdings, increase in older vehicle sales, and postponing vehicle purchase *(12, 13)*. Additional studies highlight the positive impact of carsharing on VMT *(14, 15)*.

To the best of our knowledge, this study in the first to focus on shared ownership and use of autonomous vehicles with an analytical formulation. In order to implement the shared ownership program, we need to form clusters of households, where members of each cluster jointly own a set of autonomous vehicles. The goal is to increase efficiency by finding the minimum number of vehicles each cluster requires, and allowing members of each cluster to rideshare if the opportunity presents itself. This is similar to the problem of finding the minimum number of vehicles in a dial-a-ride problem (DARP) with time windows *(16, 17)*.

In addition, we allow clusters to rent out their vehicles using a central carsharing system. The problem of allocating vehicles to dynamic requests is similar to the dynamic DARP with time windows.

Cordeau et al. *(18)* provides a literature review on the algorithms developed for the dynamic DARP. The computational time, and/or number of requests these algorithms are able to manage presents significant limitations for the purpose of this paper. We propose a 'greedy' heuristic algorithm that is able to provide high quality solutions within a short period of time.

Finally, we implement the shared vehicle ownership and use program for a sample of households in San Diego, California, and comment on the resulting efficiency.

SHARED VEHICLE OWNERSHIP AND USE PROGRAM

Envision a set of households F, who share the ownership of a set of autonomous vehicles V. These households form a cluster to which the vehicles under their shared ownership belong. Each vehicle $v \in V$ has the capacity to carry C_v number of passengers. Each household $f \in F$ has a set of essential trips that need to be served by the set of autonomous vehicles. We define set M_e to include all the essential trips of the households that belong to the cluster. Common types of essential trips may include work-based trips, grocery shopping, and trips to school. However, households can include any type of trips in the set of essential trips, if they need to ensure regular access to vehicles for such trips.

For a given trip k, a cluster member needs to input into the system the location of the origin of the trip, OS_k, the location of the destination of the trip, DS_k, the earliest departure time from the origin location, ED_k, and the latest arrival time at the destination location, LA_k.

While vehicles are idle, they can be rented out to satisfy a set of on-demand transportation requests, M, in order to cover a part of the system cost. A rental request $k \in M$ should include the location where a vehicle needs to deliver itself (OS_k), and the location where it needs to return (DS_k), along with the rental period duration (P_k), and the rental time window ($[ED_k, LA_k]$).

The first goal of the system is to advise households in a cluster on the minimum number of vehicles they need to purchase to cover their set of essential trips. In the interest of higher efficiency, the system is designed to allow cluster members to rideshare, if the spatiotemporal proximity of their trips permit it. The second goal of the system is to maximize the total number of on-demand carsharing requests, in order to maximize the external revenue generated. These goals are implemented sequentially, i.e. we first determine the minimum number of vehicles for each cluster of households, and then use these vehicles for carsharing during their idle times. In the next section, we mathematically model these two stages.

Mathematical Modeling

In order to model the system defined in the previous section, we formulate two optimization problems. The first problem finds the minimum number of autonomous vehicles that should be owned by a cluster, in order to guarantee that its set of essential trips will be served, and provides vehicle itineraries. The second problem uses the vehicles' idle times to serve the maximum number of on-demand carsharing requests.

To formulate these two problems, we need to first define a number of sets. For a given cluster, we define a set of stations, S_e, that contains the origin and destination locations of the cluster's essential trips. Furthermore, we define set S to contain all the origin and destination locations of all essential and non-essential trips (by all clusters). By introducing stations, we discretize the space dimension of the problem. In addition, we discretize the study time horizon into a short time intervals, dt. We define set T to include all time intervals in the study time horizon. In this study, we use $dt = 5$ min. In a network discretized in both time and space, we define a node n as a tuple $(t_i, s_i) \in T \times S$. Consequently, we define a link ℓ as a tuple of nodes $\ell = (n_i, n_j) = (t_i, s_i, t_j, s_j)$, where $(t_j - t_i)dt$ is the travel time between stations i and j. We define set L to include all links.

Furthermore, we define an origin depot, D_o, and a destination depot D_d. The depot stations are not real locations on the network, and are used to assist in the formulation of the problem. D_o is connected to all stations in set S, and all stations in S are connected to D_d. Furthermore, D_o and D_d are connected to each other. Figure 2 displays a typical network and demonstrates the connection between the depots and set of stations.

We perform a pre-processing procedure to reduce the size of the input sets to the problem *(19)*. Through this procedure, we find the links that are spatiotemporally reachable for each trip $k \in \{M \cup M_e\}$ given its time window, and keep such links in set L_k. Therefore, when formulating the problem, we don't need to place explicit constraints on the time windows of trips, since only links with feasible time windows are members of set L_k.

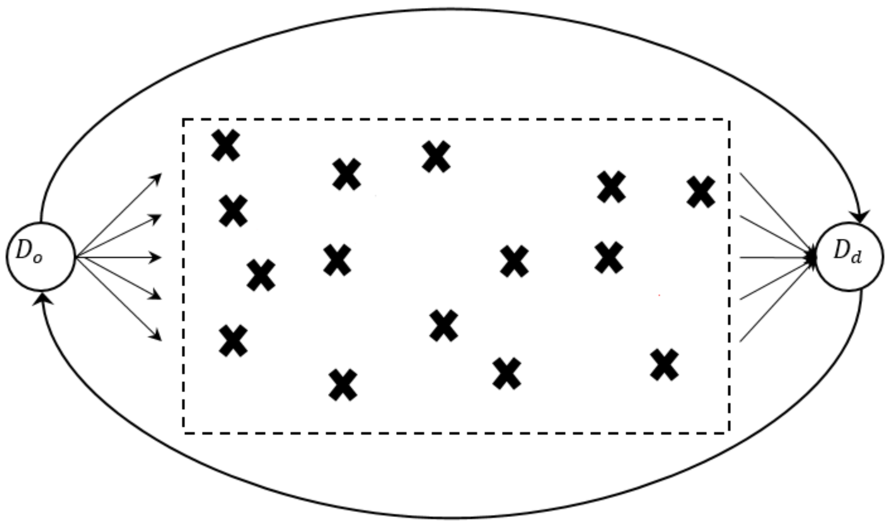


FIGURE 2 A typical network to demonstrate the connection of depot stations together, and to members of set *S*

Routing of Autonomous Vehicles

The problem of finding the minimum number of vehicles required to serve a cluster's set of essential trips is formulated through equations $(3) - (10)$. The formulation requires two sets of decision variables defined in (1) and (2).

$$x_\ell^v = \begin{cases} 1 & \text{If vehicle } v \text{ travels on link } \ell \\ 0 & \text{Otherwise} \end{cases} \tag{1}$$

$$y_\ell^{kv} = \begin{cases} 1 & \text{If trip } k \text{ is carried out by vehicle } v \text{ on link } \ell \\ 0 & \text{Otherwise} \end{cases} \tag{2}$$

Given that the households in a cluster have a total of m members, $|m|$ is an upper-bound on the number of vehicles needed to serve the cluster. Therefore, to determine the minimum number of vehicles required, we formulate an optimization problem, assuming there to be $|m|$ vehicles available, and try to maximize the number of vehicles that are not used.

Constraint sets (4) and (5) force all vehicles to go back from D_d to D_o at the end of the day. Constraint set (6) is the flow conservation constraint, forcing all vehicles that enter a station at a given time interval to exit that station at the same time interval. Notice that vehicles do not have to physically leave a station. Members of set L in the form $\ell = (t, s, t + 1, s)$ can cover such situations, where a vehicle can stay at a station for one time interval.

Constraint sets $(7) - (9)$ route the set of trips in the network. Constraint set (7) and (8) ensure that a trip leaves its origin station and enters its destination station within the trip's time window, respectively. Constraint set (9) is the flow conservation constraint.

Constraint set (10) serves two purposes: it links vehicle routes with trip routes, and ensures that the capacity of vehicles is not exceeded.

Vehicles that are excessive and are not actually routed in the system have to take the link that connects D_o to D_d. Therefore, to minimize the number of used vehicles, we maximize the vehicles that travel on this link, as mathematically stated in the objective function of the problem in (3). The second term in the objective function minimizes the total travel time by vehicles in the network. We set a negative weight W for the first term in the objective function to take into account the relative importance of minimizing number of vehicles in a cluster compared to the total travel time by the cluster members in the network.

The solution to this problem simultaneously provides the minimum number of vehicles required to serve the essential trips, and itineraries for trips and the vehicles.

Minimize
$$W \sum_{\substack{v\in V,\ell\in L:\\ s_i=D_o,s_j=D_d}} x_\ell^v + \sum_{\substack{v\in V,\ell\in L:\\ s_i\neq s_j}} (t_j - t_i) x_\ell^v \tag{3}$$

Subject to:
$$\sum_{\substack{\ell\in L:\\ s_i=D_d,s_j\in S_e-D_o}} x_\ell^v = 0 \qquad \forall v \in V \tag{4}$$

$$\sum_{\substack{\ell\in L:\\ s_i=D_d,s_j\in D_o}} x_\ell^v = 1 \qquad \forall v \in V \tag{5}$$

$$\sum_{\substack{s_i\in S_e,t_i\in T:\\ \ell=(t_i,s_i,t,s)\in L}} x_\ell^v = \sum_{\substack{s_j\in S_e,t_j\in T:\\ \ell=(t,s,t_j,s_j)\in L}} x_\ell^v \qquad \forall v \in V, t \in T, s \in S_e - D_o : (t_i, s_i, t, s) \in L \tag{6}$$

$$\sum_{v\in V} \sum_{\substack{\ell\in L_k:\\ s_i=OS_k}} y_\ell^{kv} - \sum_{v\in V} \sum_{\substack{\ell\in L_k:\\ s_j=OS_k}} y_\ell^{kv} = 1 \qquad \forall k \in M_e \tag{7}$$

$$\sum_{v\in V} \sum_{\substack{\ell\in L_k:\\ s_j=DS_k}} y_\ell^{kv} = 1 \qquad \forall k \in M_e \tag{8}$$

$$\sum_{\substack{s_i\in S_e,t_i\in T:\\ \ell=(t_i,s_i,t,s)\in L_k}} y_\ell^{kv} = \sum_{\substack{s_j\in S_e,t_j\in T:\\ \ell=(t,s,t_j,s_j)\in L_k}} y_\ell^{kv} \qquad \forall v \in V, k \in M_e, t \in T, s \in S_e - \{OS_k, DS_k\} : (t_i, s_i, t, s) \in L_k \tag{9}$$

$$\sum_{k\in M_e:\ell\in L_k} y_\ell^{kv} \leq x_\ell^v \qquad \forall v \in V, \ell \in L \tag{10}$$

On-demand Carsharing

The pick-up and drop-off schedules for the set of essential trips are determined by the optimization problem in the previous section. The idle times of vehicles can be used to serve on-demand transportation requests through a carsharing program.

Autonomous vehicles become free after dropping off a passenger, and before picking up the next. During this period, a vehicle needs to make a trip from the destination location of the first passenger, to the origin location of the second, in a travel time window that is bounded from below by the scheduled arrival time of the first passenger, and from above by the scheduled departure time of the second. The first optimization problem ensures that this travel time window is larger than the actual travel time between the locations. Although this time window is not strictly an idle period, the vehicle can use the extra time to serve on-demand carsharing requests, and that is why we refer the time windows between two consecutive scheduled pick-up and drop-offs as free travel time window.

In order to mathematically formulate the carsharing problem, we identify the set of free time windows between scheduled trips, and try to find the maximum number of carsharing requests that can be satisfied during these time windows. We keep the set of free time windows for each vehicle $v \in V$ in set $J(v)$. The j^{th} free time window of autonomous vehicle v starts after dropping off its j^{th} assigned passenger, and ends when passenger $j+1$ needs to be picked up. We denote this travel time window by $[ED_{v,j} \;\; LA_{v,j}]$. During this time window, the vehicle needs to travel from the destination location of its j^{th} scheduled trip, to the origin location of its next scheduled trips. We denote these parameters by $OS_{v,j}$ and $DS_{v,j}$ respectively.

We formulate this problem using three sets of decision variables in equations $(11)-(13)$.

$$x_\ell^{vj} = \begin{cases} 1 & \text{If vehicle } v \text{ travels on link } \ell \text{ during its } j^{th} \text{ free time window} \\ 0 & \text{Otherwise} \end{cases} \tag{11}$$

$$y_\ell^{kvj} = \begin{cases} 1 & \text{If request } k \text{ is served through the } j^{th} \text{ free time window of vehicle } v \text{ on link } \ell \\ 0 & \text{Otherwise} \end{cases} \tag{12}$$

$$z_k = \begin{cases} 1 & \text{If carsharing request } k \text{ is served} \\ 0 & \text{Otherwise} \end{cases} \tag{13}$$

Contrary to the problem in $(11)-(13)$ where all vehicles had the same link set L, here each vehicle has a different link set in each of its free time windows. Let us keep the set of links for vehicle v during its j^{th} free window in set L_{vj}. Furthermore, we introduce a new set $L_{kvj} = \{L_k \cap L_{vj}\}$. This set includes all the links that are accessible to both vehicle v during its j^{th} time window, and request k.

The constraint sets that defines this problem are very similar to constraint sets in the previous section, where we routed the autonomous vehicles to satisfy the set of essential trips for each cluster.

Constraint sets $(15)-(17)$ route vehicles within their free time windows. Constraint set (15) ensures that each vehicle at each of its free time windows leaves its origin station after delivering its last passenger. Constraint set (16) ensures that the vehicle reaches its destination location before the departure time of its next scheduled passenger. Constraint set (17) is the flow conservation constraint.

Constraint sets $(18)-(20)$ route on-demand requests in the network. These sets of constraints are similar to constraint sets $(15)-(17)$ that route vehicles, with a small variation that not all on-demand requests can be necessarily served. This is reflected in the formulation by replacing 1 on the right hand side of constraint sets (15) and (16) by variable z_k in constraint sets (18) and (19).

Finally, constraint set (21) ensures that each served request is assigned a single vehicle, and each vehicle is assigned to only one request at a time. The objective of the carsharing problem (14) is to maximize the total number of served requests.

$$\text{Minimize} \quad \sum_{k \in M} z_k \tag{14}$$

$$\text{Subject to:} \quad \sum_{\substack{\ell \in L_{vj}: \\ s_i = OS_{v,j}}} x_\ell^{vj} - \sum_{\substack{\ell \in L_{vj}: \\ s_j = OS_{v,j}}} x_\ell^{vj} = 1 \qquad \forall v \in V, j \in J(v) \tag{15}$$

$$\sum_{\substack{\ell \in L_{vj}: \\ s_j = DS_{v,j}}} x_\ell^{vj} = 1 \qquad \forall v \in V, j \in J(v) \tag{16}$$

$$\sum_{\substack{s_i \in S, t_i \in T: \\ \ell=(t_i,s_i,t,s) \in L_{vj}}} x_\ell^{vj} = \sum_{\substack{s_j \in S, t_j \in T: \\ \ell=(t,s,t_j,s_j) \in L_{vj}}} x_\ell^{vj} \qquad \forall v \in V, j \in J(v), \; \forall\, t \in T, s \in S - D_o \; : (t_i, s_i, t, s) \in L_{vj} \tag{17}$$

$$\sum_{\substack{\ell \in L_{kvj}: \\ s_i = OS_k}} y_\ell^{kvj} - \sum_{\substack{\ell \in L_{kvj}: \\ s_j = OS_k}} y_\ell^{kvj} = z_k \qquad \forall v \in V, j \in J(v), \; \forall k \in M \tag{18}$$

$$\sum_{\substack{\ell \in L_{kvj}: \\ s_j = DS_k}} y_\ell^{kvj} = z_k \qquad \forall v \in V, j \in J(v), \; k \in M \tag{19}$$

$$\sum_{\substack{s_i \in S, t_i \in T: \\ \ell=(t_i,s_i,t,s) \in L_{kvj}}} y_\ell^{kvj} = \sum_{\substack{s_j \in S, t_j \in T: \\ \ell=(t,s,t_j,s_j) \in L_{kvj}}} y_\ell^{kvj} \qquad \forall v \in V, j \in J(v), k \in M, \; t \in T, s \in S - \{OS_k, DS_k\} \; : (t_i, s_i, t, s) \in L_{kvj} \tag{20}$$

$$\sum_{k \in M: \ell \in L_{kvj}} y_\ell^{kvj} \leq x_\ell^{vj} \qquad \forall v \in V, j \in J(v), \ell \in L_{vj} \tag{21}$$

SOLUTION METHOD

We formulated the first optimization problem to find the minimum number of autonomous vehicles required to serve a cluster's set of essential trips, and optimally route these vehicles. This problem does not need be solved in real-time, and therefore for problems of moderate size (as we will show later) optimization engines such as CPLEX can be used to solve it.

The second optimization problem that maximizes the number of carsharing requests may need to be solved in real-time, as on-demand carsharing requests arrive. In this section, we devise a greedy heuristic algorithm to solve this problem in real-time. The numerical study that follows illustrates the level of efficiency and accuracy of this heuristic algorithm.

The carsharing problem as described in the previous section bears similarities to the family of parallel machine scheduling problems in manufacturing. This class of problems includes a large variety of problems, and is used to find the optimal sequence of using machinery in manufacturing processes.

Parallel machine scheduling problems vary in job characteristics (whether there are preemptive or precedence constraints present, fixed/relaxed start or finish time, etc.), machine characteristics (identical or non-identical, serial or parallel, etc.), and the optimality criteria (max number of completed jobs, min makespan, etc.). In the context of our carsharing problem, jobs are carsharing requests, and machinery are the free time windows of drivers. The problem we are trying to solve has the following characteristics:

(1) No preemptive or precedence constraints present: Once we fix the schedules of the essential trips, the vehicles' free time windows can be used in any manner, i.e. there is no precedence requirement on the sequence of the carsharing request to be satisfied.

(2) Multiple non-homogeneous machines/servers: In our problem each free time window of each vehicle acts as a separate server. Furthermore, our servers are non-homogeneous, meaning that each vehicle at each of its free time windows has distinct origin and station, as well as start and finish times.

(3) Jobs are available during specified time windows, rather than with specific start and finish times: the carsharing requests specify a time window during which a vehicle is required, rather than specify the exact time for start and end of their requests.

(4) Set-up cost: In our problem there exist server- and job sequence-dependent set-up costs. Because vehicles have to travel to location where they are requested, which vehicle is to be assigned to a request, and the sequence of requests assigned to a vehicle, all play a role.

(5) Objective: maximizing the number of served jobs (satisfied carsharing requests).

There is an extensive amount of literature on machine scheduling *(20, 21)*. Rabadi et al. *(22)* propose heuristics to solve the non-preemptive unrelated parallel machine scheduling problem, in which machine- and job sequence-dependent setup times are considered, but jobs are all assumed to be available at time zero. Gabrel *(23)* proposes heuristics to solve the problem of scheduling non-preemptive jobs with an interval for starting time, on identical parallel machines. To the best of our knowledge, there is no study that combines both characteristics (set-up costs, and time windows for jobs), that can be used to solve the carsharing problem formulated in the previous section.

Heuristic Algorithm to Solve the Carsharing Problem

The heuristic algorithm described in this section is based on the earliest finishing time (EFT) heuristic originally designed to solve the interval scheduling problem. In the interval scheduling problem, there is a machine that needs to complete the maximum number of jobs possible. Each job has a specific start and finish time. At each step, the EFT heuristic selects the job with the earliest finishing time that does not conflict with the previously selected jobs. The EFT heuristic yields optimal solutions.

The carsharing problem we need to solve is substantially more complicated than the interval scheduling problem. In fact, it is easy to see that the carsharing problem is NP-Hard. Here, we modify the EFT heuristic, and tailor it to solve the carsharing problem. Our proposed algorithm is displayed in Figure 3(a).

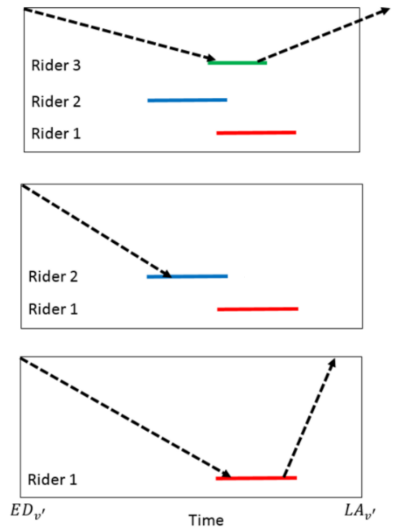
In the mathematical program in $(14)-(21)$, we used the tuple (v, j) to refer to the j^{th} free time window of vehicle v. In the interest of simplifying notation, we treat each free time window of each vehicle as a separate vehicle $v' \in V'$, where $V' = \{(v, j) | v \in V, j \in J(v)\}$.

Algorithm: on-demand vehicle allocation

1. Initialize: $\forall v' \in V'$
   $Loc(v') = OS_{v'}$
   $Time(v') = ED_{v'}$
2. Find the set of feasible requests $R(v')$, $\forall v' \in V'$
   $\forall k \in R$:
   If $Max\{Time(v') + shp^1(Loc(v'), OS_k), ED_k\} + shp(OS_k, DS_k) \leq LA_k$
   $R(v') = R(v') \cup \{k\}$
3. Find the matched request and driver by studying the minimum finishing time for all combinations of vehicles and requests $(\forall v' \in V', k \in R(v'))$
   $(v'^*, k^*) = Argmin_{v' \in V', k \in R(v')}\{Max\{Time(v') + shp(Loc(v'), OS_k), ED_k\} + shp(OS_k, DS_k)\}$
4. Update sets
   $Loc(v'^*) = DS_{k^*}$
   $Time(v'^*) = Max\{Time(v'^*) + shp(Loc(v'^*), OS_{k^*}), ED_{k^*}\} + shp(OS_{k^*}, DS_{k^*})$
5. Delete k^* from $R(v'), \forall v' \in V'$.
6. Update $R(v'^*)$ based on rider travel time windows:
   $\forall k \in R(v'^*)$:
   If $Max\{Time(v'^*) + shp(Loc(v'^*), OS_k), ED_k\} + shp(OS_k, DS_k) > LA_k\}$
   $R(v'^*) = R(v'^*) \backslash k$
   Go to step 3.
7. Stopping Criteria: $\forall v' \in V', R(v') = \emptyset$.

[1] $shp(i, j)$: shortest path travel time between i and j

(a) Heuristic algorithm to solve the carsharing problem



(b) Determining the set of feasible requests $R(v')$ for vehicle v'

FIGURE 3 Greedy heuristic algorithm to solve the carsharing problem

In the first step of the algorithm, we initialize two sets of arrays. The first array, $Loc(v')$, determines the current location of vehicle v'. The second array, $Time(v')$, determines the time vehicle v' becomes idle (available). We initialize the location array Loc for each vehicle $v' \in V'$ with the origin station of the vehicle, and the time array $Time$ with the earliest departure time of the vehicle.

The algorithm starts by determining the set of feasible carsharing requests for each vehicle. In order for a request to be feasible for a vehicle, the vehicle should be able to drive from its current location to the request's origin station, and get there at or after the start of the request's time window, stay in possession of the requester for the requested duration of time, and finally arrive at its own destination (the pick-up location of its next scheduled essential trip) before its latest arrival time. Figure 3(b) studies the feasibility of three carsharing requests for a vehicle. The boundaries of the boxes show the free time window of the vehicle, and the line (blue, red, green) associated with each request marks its time window. The first request (at the bottom) is feasible for the vehicle. The vehicle arrives at the request's origin location after the request's earliest departure time, is able to stay in possession of the requester for the demanded duration that ends before the request's latest arrival time, and travels to its destination station within its time window. The second and third requests, however, are not feasible for the vehicle. In the case of the second request, the vehicle cannot stay in possession of the requester for the duration of the request, and in the case of the third request, the vehicle cannot go back to its own destination station after finishing serving the request.

In the third step of the algorithm, we find finishing times for all combinations of vehicles and their set of feasible requests. The finishing time of vehicle v' serving request k includes the time required for the vehicle to arrive at the request's origin location, and then stay in possession of the requestor for the demanded period of time. Note that if the vehicle arrives at a requested location before the start of the request's time window, it has to wait until the start of the time window. The vehicle and request pair that lead to the earliest finishing time will be selected and matched together.

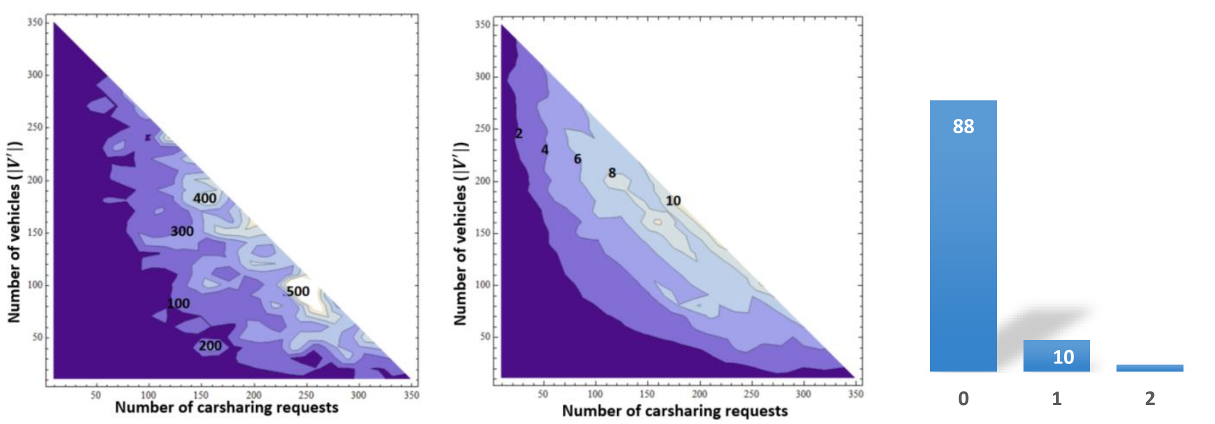
In step 4, the location of the matched vehicle will be updated to the location of the destination of the matched request, and the time array of the assigned vehicle will be updated to the drop-off time of the rented out vehicle.

In step 5, the matched request in step 3 will be eliminated from the set of available requests to all vehicles. Furthermore, since the time window and location of the matched vehicle in step 3 have been updated, the set of feasible requests for this vehicle needs to be updated as well. The algorithm stops when all vehicles have empty sets of feasible requests.

Heuristic Algorithm Accuracy and Efficiency

To evaluate the performance and efficiency of the proposed heuristic algorithm, we generated 380 random instances of the carsharing problem, each in a randomly-generated grid network. The number of vehicles, $|V'|$, and on-demand requests (|M|) in the problem instances vary between 20 and 350. We solved the problem instances both directly (by solving the carsharing optimization problem using AMPL CPLEX), and by means of the heuristic algorithm, on a PC with Core i7 3 GHz and 8GB of RAM. The solution times are reported in figure 4. This figure suggests that the savings in solution times are substantial.

Next, we compared the number of served riders using the heuristic algorithm to the optimal solution, in order to assess the performance of the algorithm. The total number of served requests in all 380 randomly generated problems solved by the heuristic algorithm was within 98% of the total number of served requests obtained by solving the problems to optimality using the AMPL CPLEX optimization engine. Figure 4(c) demonstrates the performance of the heuristic algorithm in more detail. This figure suggests that in 88% of the problem instances, we were able to obtain the optimal solution using the heuristic algorithm. In 10% of the problem instances, the heuristic solution managed to serve one less request than the optimal solution, and in 2% of the problem instances the heuristic solution was outweighed by the optimal solution by 2 served requests. The numerical tests imply that the trade-off between the performance and solution time, that is inevitable when replacing an optimal algorithm with a heuristic, may be in favor of the heuristic algorithm.



(a) Solution time contour plot (sec) for solving the optimization problem using AMPL CPLEX

(b) Solution time contour plot (sec) for solving the carsharing instances using the heuristic algorithm

(c) Distribution of deviation of the heuristic solution from the optimal solution

FIGURE 4 Performance of the carsharing heuristic algorithm

REAL-LIFE APPLICATION

We implemented the “shared ownership and use” program for a sample of households in San Diego, using data from the 2000-2001 California Statewide household travel survey. Using this survey, Caltrans managed to collect travel data from 17049 volunteer household from throughout California. Although the number of households completing the survey was less than 0.1 percent of the total number of household residing in California in 2000, the sample was determined by the survey conductors to be a good representation of the state population.

After cleaning the data by eliminating records with incomplete or faulty information, a total of 1184 households residing in the San Diego County were retained. For these households, information on the number of household members, logged trips of each member during a working day along with the purpose of each trip, and the number of vehicles owned by households were available, among other information.

We determined the set of essential trips for each household based on the information on the purpose of the trips. We categorized trips concerning work, school, childcare, medical, fitness, community meetings, volunteer activities, visiting friends and family, and entertainment activities as essential, and the rest of the trips as non-essential. Among the 1184 households residing in the San Diego County, 573 of them did not report any essential trips during the survey day, and therefore were not considered for the shared ownership program. These households, however, were taken into consideration for the carsharing program (which will be discussed later). The remaining 611 households were considered for clustering.

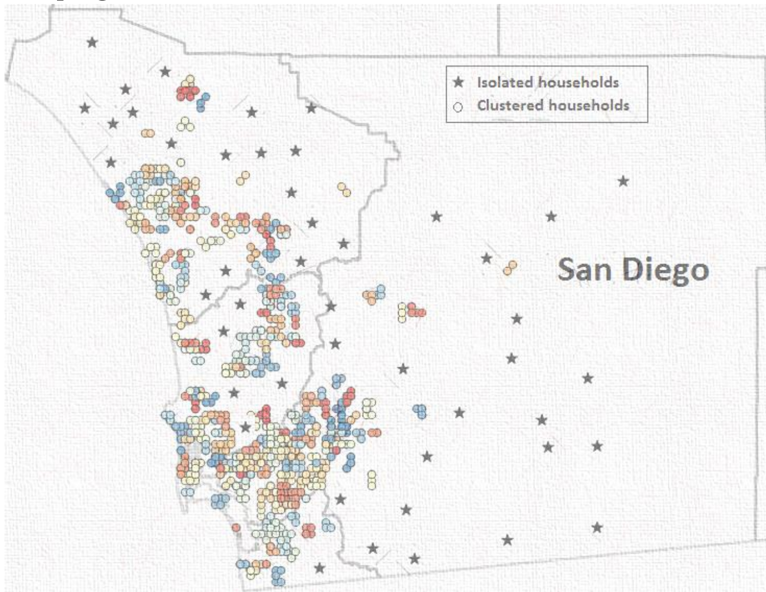
The first step in implementing the program is to cluster households. Each cluster should include a number of households with enough commonalities that would interest them to participate in the shared vehicle ownership and use program together. Various parameters can be used to determine a suitable cluster for a households, including home location, demographics and social status of household members, level of spatiotemporal proximity of trips between households, and income level, to name a few. In the interest of simplicity, we determine the clusters solely based on household home locations, using the hierarchical agglomerative clustering method *(24)*.

Figure 5(a) displays the resulting clusters, distinguished by color. About 30% of the households were geographically isolated from others, and therefore remained as stand-alone clusters. The remaining 428 households formed a total of 277 clusters. Figure 5(b) shows the distribution of number of households in clusters. A considerable number of households remain stand-alone. This result is in fact expected, given the small sample size. It is expected that considering the full population, more households would be interested in participating in such a program.

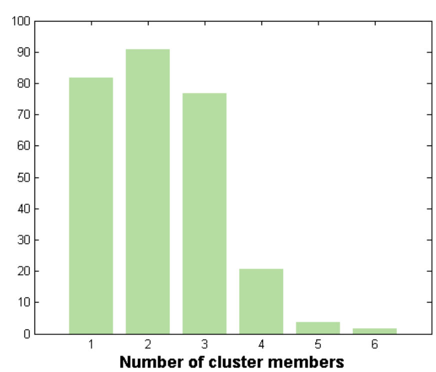
For each cluster, we solved the optimization problem in $(3) - (10)$ to find the minimum number of vehicles required to cover the cluster's set of essential trips. Solution times are displayed in figure 5(c). Not surprisingly, average solution times increase with cluster size. However, for cluster size of 6, which is the largest cluster size, the average solution time remains around 300 seconds, which is encouraging. The solution suggests that a total of 379 vehicles are required to serve all the essential trips by all clusters (including the one-household clusters). Comparing this number to the 1231 number of vehicles owned by the 611 households with non-empty sets of essential trips suggests that this program has the potential to have significant impacts on lowering vehicle ownership. Note that the 379 vehicles are calculated only based on households' set of essential trips, and all the 1184 households in our sample still need to make their non-essential trips. Therefore, this number serves as a lower bound to the number of required vehicles. Later, we will compute the number of additional vehicles required to serve the non-essential trips.

Figure 6(a) shows the distribution of number of vehicles in clusters. This figure shows that about 200 clusters need no more than one vehicle to serve their essential trips. No cluster needs more than 4 vehicles. Figure 6(b) shows the distribution of number of vehicles households in each cluster owned in year 2000. This figure suggests that a large proportion of households owned at least 2 vehicles.

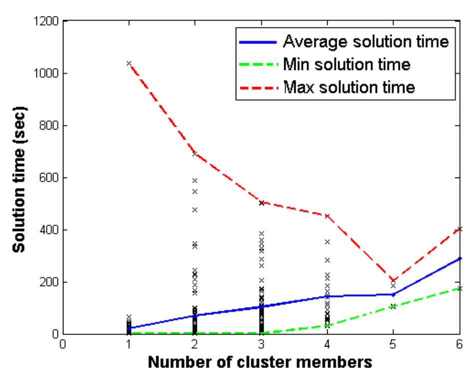
After forming clusters of households, and routing autonomous vehicles to serve clusters' sets of essential trips, we need to address the non-essential trips. One possibility is to use the carsharing program for this purpose. The question is, what percentage of the non-essential trips can be served using the vehicles owned by clusters of households during their idle times. We used our heuristic algorithm to rent out the 379 vehicles with the goal of serving as many non-essential trips as possible. We managed to serve 63% percent of the non-essential trips. Using the on-demand vehicle allocation algorithm in figure 3(a), we need 125 additional vehicles to serve the entire set of non-essential trips. This brings the total number of vehicles required to 504. In the year 2000, the 1184 households in our sample owned 2194 vehicles. Therefore, this program results in a more than 4-fold reduction in vehicle ownership.



(a) Clusters of households. Households in each cluster are assumed to share ownership of a set of autonomous vehicles



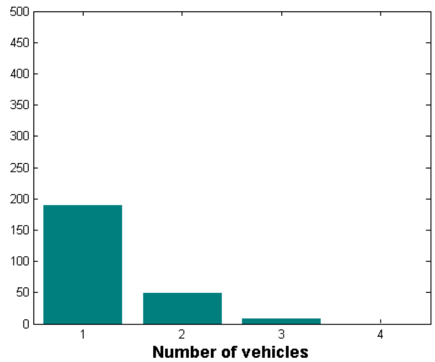
(b) Distribution of clusters based on number of households in them



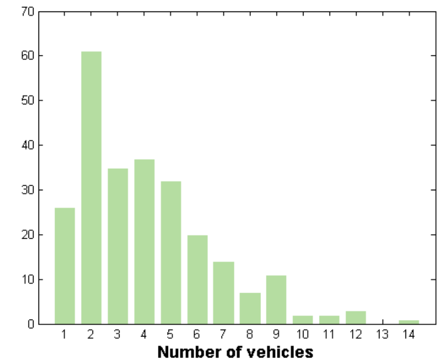
(c) Solution time (sec) of finding the optimal number of vehicles and vehicle itineraries for each cluster size

FIGURE 5 Clusters of households in San Diego, California

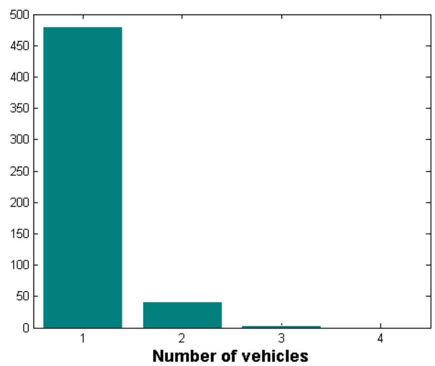
The savings in the number of vehicles in the proposed system originate from three different sources: 1- introduction of autonomous vehicles, 2- shared ownership of vehicles, and 3- ridesharing. It would be interesting to see how much of the savings can be attributed to each source. Towards this goal, we consider two additional cases. In the first case, we study the impact of households trading their current vehicles for autonomous vehicles. We optimize the number of autonomous vehicles required for each household. Results are displayed in figure 6(c). Figure 6(d) displays the current number of (ordinary) vehicles households own. The total number of vehicles for all households changes from 2194 to 787, a near 2.5-fold decrease.



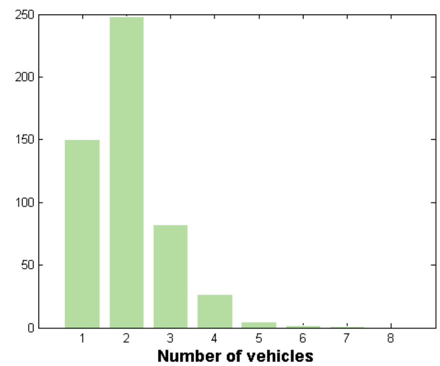
(a) Histogram of number of autonomous vehicles required for clusters, considering ridesharing



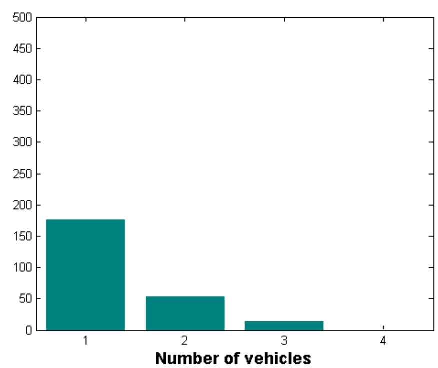
(b) Histogram of number of vehicles currently owned by cluster members



(c) Histogram of number of autonomous vehicles required for individual households



(d) Histogram of number of vehicles currently owned by individual households



(e) Histogram of number of autonomous vehicles required by households, if participating in the shared ownership, but not shared-use program

FIGURE 6 Vehicle Ownership Effects of Implementing the shared ownership and use program in the city of San Diego

In the second case, we consider only a shared ownership system, without the possibility of ridesharing. The results are displayed in figure 6(e). The total number of autonomous vehicles required in this case decreases to 528, a decrease of about 33% with respect to the previous case where the autonomous vehicles were owned by individual households rather than being shared. This implies that the shared ownership component of the program has a considerable impact on the reduction of vehicle ownership.

Adding ridesharing to the mix (Figure 6(a)) decreases the number of vehicles from 528 to 504, a decrease of about 5%, which although may not seem substantial, but can increase exponentially as more households decide to participate in the shared ownership and use program, and the cluster sizes increase. Comparing figures 6(a) and 6(e), it is also interesting to notice that introducing ridesharing, in addition to decreasing the total number of required vehicles, changes the distribution of vehicles by increasing the number of one-vehicle clusters.

Although the sample of household used in this study was too small, the implementation of the shared vehicle ownership and use program led to substantial increase in efficiency. It can only be expected that with more households interested in participating in the program, the efficiency is going to increase at a higher rate.

CONCLUSION

In this paper, we propose a model for shared ownership and use of autonomous vehicles. The motivation behind this model was to encourage the population to switch to autonomous vehicles, by lowering the cost of autonomous vehicle ownership through a shared vehicle ownership and use program. For a group of households willing to participate in the program together, we formulated an optimization problem to find the minimum number of vehicles required to satisfy their transportation needs. Participants in the program can register their vehicles in a central carsharing program when they are not being used, in order to generate revenue.

We implemented this program for a sample of households in San Diego, with 1184 households. We formed clusters of households based on proximity of home locations. The total number of vehicles required to cover the transportation demand of the entire sample under the shared ownership and use program experienced a 4-fold reduction.

REFERENCES

1. Curvy Road, http://www.curvyroad.com/, Accessed July 22, 2015.
2. Schall, E. *The Future of Car Ownership and Autonomous Driving,* Sep. 30, 2015, http://www.cheatsheet.com/automobiles/the-future-of-car-ownership-and-autonomous-driving.html/?a=viewall, Accessed Aug. 1 2015,
3. Naughton K. Driverless*, Shared Cars to Cut Vehicle Ownership by Half: Barclays,* May 19, 2015. http://www.insurancejournal.com/news/national/2015/05/19/368711.htm, Accessed Aug. 1, 2015.
4. Fagnant, D. J., and K. M. Kockelman. The travel and environmental implications of shared autonomous vehicles, using agent-based model scenarios. *Transportation Research Part C: Emerging Technologies*, Vol. 40, 2014, pp. 1-13.
5. Schoettle, B., and M. Sivak. Potential Impact of Self-Driving Vehicles on Household Vehicle Demand and Usage. UMTRI-2015-3. University of Michigan, Transportation Research Institute. Feb. 2015.
6. Garigliano J., *Own a Lamborghini – part time,* March 2, 2007, http://money.cnn.com/magazines/fsb/fsb_archive/2007/02/01/8399941, Accessed July 22, 2015.
7. John, S., and S. Phadnis. *Ford to pilot car-sharing project in Bengaluru*. http://timesofindia.indiatimes.com/business/india-business/Ford-to-pilot-car-sharing-project-in-Bengaluru/articleshow/45802696.cms. Jan. 8, 2015, Accessed July 22, 2015.
8. ZoomCar Facebook Page. *Your chance to redefine urban mobility with Ford and Zoomcar*. https://www.facebook.com/zoomcar.in/posts/533737686763146?comment_id=534786236658291&re

ply_comment_id=534938423309739&total_comments=1&comment_tracking=%7B%22tn%22%3A%22R9%22%7D, Jan. 19, 2015, Accessed July 22, 2015.

9. Chan, N. D., and S. A. Shaheen. Ridesharing in North America: Past, present, and future. *Transport Reviews*, Vol. 32, No.1, 2012, pp. 93-112.
10. Heinrich, S. *Implementing real-time ridesharing in the San Francisco Bay Area*. Ph. D. thesis, San Jose State University. 2010.
11. Barth, M., and S. A. Shaheen. Shared-use vehicle systems: Framework for classifying carsharing, station cars, and combined approaches. *Transportation Research Record: Journal of the Transportation Research Board*1791, 2002, pp. 105-112.
12. Martin, E., S. A. Shaheen, and J. Lidicker. Impact of carsharing on household vehicle holdings: Results from North American shared-use vehicle survey. *Transportation Research Record: Journal of the Transportation Research Board* 2143, 2010, pp. 150-158.
13. Walb, C., and W. Loudon. *Evaluation of the Short-Term Auto Rental (STAR) Service in San Francisco, CA*. US Department of Transportation, Urban Mass Transportation Administration, 1986.
14. Cervero, R., and Y. Tsai. City CarShare in San Francisco, California: second-year travel demand and car ownership impacts. *Transportation Research Record: Journal of the Transportation Research Board* 1887, 2004, pp. 117-127.
15. Cervero, R., A. Golub, and B. Nee. City CarShare: longer-term travel demand and car ownership impacts. *Transportation Research Record: Journal of the Transportation Research Board* 19922007, pp. 70-80.
16. Jang-Jei, J., A. R. Odoni, H. N. Psaraftis, and N. H. M. Wislon. A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. *Transportation Research Part B: Methodological*. Vol. 20, No. 3, 1986, pp. 243-257.
17. Psaraftis, H. N. An exact algorithm for the single vehicle many-to-many dial-a-ride problem with time windows. *Transportation Science*. Vol. 17, No. 3, 1983, pp. 351-357.
18. Cordeau, J. F., and G. Laporte. The dial-a-ride problem: models and algorithms. *Annals of Operations Research*. Vol. 153, No. 1, 2007, pp. 29-46.
19. Masoud, N., and R. Jayakrishnan. "A decomposition Algorithm to solve the multi-hop peer-to-peer ride-matching problem." *Transportation Research Board 94th Annual Meeting*. No. 15-2755. 2015.
20. Hall, N. G. and C. Sriskandarajah. A survey of machine scheduling problems with blocking and no-wait in process. *Operations Research*, Vol. 44, No. 3, 1996, pp. 510–525.
21. Cheng, T., Q. Ding, and B. Lin. A concise survey of scheduling with time-dependent processing times. *European Journal of Operational Research*, Vol. 152, No. 1, 2004, pp. 1– 13.
22. Rabadi, G., R. Moraga, and A. Al-Salem. Heuristics for the unrelated parallel machine scheduling problem with setup times. *Journal of Intelligent Manufacturing*, Vol, 17, No 1, 2006, pp 85–97.
23. Gabrel, V. Scheduling jobs within time windows on identical parallel machines: New model and algorithms. *European Journal of Operational Research*, Vol. 83, No. 2, 1995, pp.320 –329.
24. Steinbach, M., G. Karypis, and V. Kumar. A comparison of document clustering techniques. *KDD workshop on text mining*. Vol. 400. No. 1. 2000.