Transportation Research Part E 108 (2017) 122–140

Contents lists available at ScienceDirect

Transportation Research Part E

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

# Novel dynamic formulations for real-time ride-sharing systems

Ali Najmi*, David Rey, Taha H. Rashidi

*Research Center for Integrated Transport Innovation, School of Civil and Environmental Engineering, The University of New South Wales, Sydney 2032, Australia*

ARTICLE INFO

*Keywords:*
Ride-sharing
Dynamic
Matching policy
Rolling horizon
Clustering

ABSTRACT

This paper proposes new objective functions for the matching problem arising in ride-sharing systems based on trips' spatial attributes. Novel dynamic matching policies are then proposed to solve the problem dynamically in a rolling horizon framework. Finally, we present a new clustering heuristic to tackle instances with a large number of participants efficiently. We find that the proposed models maximize the matching rate while maintaining distance-savings at an acceptable level, which is an appealing achievement for ride-sharing systems. Further, our solution method is capable of solving large-scale instances in real-time.

## 1. Introduction

Travel cost, traffic congestion, limited capacity for car park and environmental concerns have continually encouraged people to shift their travel modes toward emerging alternatives. Ride-sharing is a promising and competitive approach to reduce private car ownership. In its original form, ride-sharing consists of picking-up riders along a trip incidental to the principal purpose of the driver. In this case, the driver intends to reach a destination and not to transport people just for profit (Code of Virginia, 1989).

It is widely acknowledged that ride-sharing may be able to meet the mobility needs of a significant share of the travellers (Stiglic et al., 2015). Smartphones have facilitated the development of ride-sharing systems by linking riders and drivers in a dynamic and on-demand environment. Hence, in modern ride-sharing systems, drivers and riders are matched automatically and both parties can be notified within short notice (Agatz et al., 2011; Gargiulo et al., 2015; Stiglic et al., 2016). Depending on the level of dynamism of the ride-sharing system, a trip notification can be sent anytime from a few hours to a few minutes before departure time.

Coordination mechanisms between drivers and riders, especially the matching problem therein, have been the focus of the research on ride-sharing models (Kamar and Horvitz, 2009). Improving system-wide and trip attributes such as the matching rate, total travel times, or total trip distances have been considerably studied in the literature. Further, for wide uptake and real-time applications, ride-sharing systems should be computationally scalable.

Due to their significant computational requirements, the majority of dynamic ride-sharing models are only applicable to small and medium scale problems, hence compromising their implementation over large metropolitan areas. For example, over 13.5 million trips are currently made in Melbourne on a daily basis. Assuming that only 0.5% of the travellers decide to opt for a ride-sharing solution, the existing solution methods are not capable of efficiently finding near-optimal solutions.

In this paper, we develop a real-time ride-sharing system that iteratively solves a matching problem in a rolling horizon approach. The matching problem solved therein is based on the formulation proposed by Agatz et al. (2011). We build on this research by proposing novel objective functions for the ride-sharing matching problem and comparing their performance against two alternative objectives. Further, we propose multiple dynamic matching policies to implement the proposed rolling horizon approach. Finally, we

* Corresponding author.
*E-mail addresses:* a.najmi@unsw.edu.au (A. Najmi), d.rey@unsw.edu.au (D. Rey), rashidi@unsw.edu.au (T.H. Rashidi).

http://dx.doi.org/10.1016/j.tre.2017.10.009
Received 2 January 2017; Received in revised form 7 September 2017; Accepted 10 October 2017
Available online 01 November 2017

propose an efficient clustering approach to decompose the announcements issued by the participants into smaller subsets and show that this heuristic approach is competitive compared to an exact solution method.

The rest of the paper is structured as follows. The literature on ride-sharing optimization models is reviewed in Section 2. Section 3 formally presents the ride-sharing problem and its mathematical formulation. Section 4 presents a rolling horizon approach to solve the problem dynamically and introduces three classes of dynamic matching policies. In Section 5, we introduce a heuristic clustering algorithm to solve the ride-sharing problem on large-scale instances. Numerical results obtained from solving realistic instances derived from Melbourne's metropolitan area are presented in Section 6. Finally, our findings and possible extensions are summarized in Section 7.

## 2. Literature review

In this section, we review the existing research on ride-sharing models. We then discuss the performance measures used for evaluating ride-sharing systems.

### 2.1. Ride-sharing models

Ride-sharing models have recently received an increasing attention in the literature. The proposed models differ in their approach to solve the optimization problem as well as in the level of input data required. Amey (2011) proposes a data-driven methodology for estimating the viability of ride-sharing at an institutional scale, the MIT campus in Cambridge, Massachusetts, USA. Given commuter-specific trip characteristics (housing location, vehicle availability, arrival/departure time and route deviation time), the author compares the potential of using ride-sharing as a travel mode based on observed trip characteristics and ridesharing patterns among commuters. In this study, the optimization problem seeks to maximize the number of matched driver-rider pairs and must decide on both the role assignment (driver or rider) to each participant and the assignment of riders to drivers. This study shows a potential system-wide vehicle miles travelled savings from 9% to up to 27%.

Agatz et al. (2011) utilize a rolling horizon solution approach to periodically optimize unmatched announcements. At each iteration of the rolling horizon, a matching problem is solved with an objective function aiming to maximize the total travel distance savings. In this model, the system is allowed to delay trip notifications until departure time. This leads to notifications often being postponed as late as possible. The authors use the Bass diffusion model (Mahajan et al., 1995) to model the adoption and the sustainability of the proposed dynamic ride-sharing system. The diffusion model contains three parameters: the total number of potential adopters, a coefficient of innovation that represents the exogenous likelihood that a new participant joins the system, and a coefficient of imitation that relates to the increase in this likelihood based on the number of participants that are already in the system. They report that when innovation and imitation rates are sufficiently high, the proposed ride-sharing system converges to a steady announcement stream in two to three weeks. As in Amey (2011), the system assigns participants' role, i.e. driver or driver.

Xing et al. (2009) introduce a dynamic ride-sharing system where drivers and riders are matched en-route. Trip preferences such as gender and smoking as well as a maximum acceptable service response time for riders are included in their model. They investigate the relationship between the number of drivers and passengers' travel time. Using a simulation-based experiment on an urban network of Bremen's metropolitan area, the authors find that increasing the number of available drivers results in higher matching rates.

Stiglic et al. (2015) design an algorithm that matches drivers and riders in a ride-sharing system with meeting points. Meeting points increase the flexibility of the ride-sharing system by expanding the set of feasible matches. They consider two objective functions in a lexicographic optimization approach: their primary objective is to maximize the total number of matches while their secondary objective aims to maximize the total travel distance savings. In line with addressing the impact of participants' flexibility, Stiglic et al. (2016) use the same model to investigate the impact of matching flexibility, detour flexibility, and scheduling flexibility on matching rates.

Ghoseiri et al. (2011) formulate a ride-sharing problem in which several constraints for vehicle occupancy, waiting time to pick up, the number of connections, detour distance for vehicles and relocation distance for passengers are considered. In addition, trip preferences based on age, gender, smoking, and pet restrictions are incorporated. As in most approaches, the authors maximize the number of matches.

In a recently published paper by Masoud et al. (2017), a peer-to-peer ride exchange mechanism is proposed to increase the matching rate and customer retention in a ride-sharing system. In the mechanism, riders have the opportunity to purchase a previously-matched rider's itinerary while the exchange of rides is accompanied with an exchange of money through the ride-sharing system. For a more comprehensive review on the state of the art of current ride-sharing systems and the existing challenges to their adoption, the reader is referred to Furuhata et al. (2013).

In this paper, we build on the existing literature and propose new objective functions for the matching sub-problem arising in the dynamic ride-sharing problem. In addition, we introduce three classes of dynamic matching policies for the rolling horizon framework that provide a range of trip notification deadlines.

### 2.2. Large-scale solution approaches

One of the main challenges of dynamic ride-sharing is to deal with a large number of participants and some heuristics have been proposed to develop scalable solutions. Shen et al. (2015) use a Filter-and-Refine framework to scale down the ride-sharing problem. In their framework, the road network is first partitioned using a gird and driver and rider requests are then filtered based on a spatio-temporal index. Pelzer et al. (2015) partition the road network into distinct regions representing certain sub-structures of the road network to reduce the solution space. In their algorithm, a match may be finalized only if the rider's destination lies on the driver's corridor. Nourinejad and Roorda (2016) use a decomposition algorithm to partition participants' announcements based on their spatial positions. The algorithm matches a driver to a rider if the origin or the destination of the rider is in the vicinity of one of the zones in which the driver's route fall.

In these proposed algorithms, the existence of a rider in the vicinity of the driver's route is the criterion used for clustering the solution space. This requires drivers' routes be periodically scanned in real-time which could pose considerable computational challenges. In this paper, we present a new clustering heuristic based on both the origin and the destination of participants, to solve the dynamic ride-sharing problem for large scale problems. Using this approach, we are able to reduce computation time by a factor of 3 while only marginally impacting solution quality.

### 2.3. System-wide ride-sharing performance measures

Multiple agents including drivers, riders, and ride-sharing providers participate in a ride-sharing system, each with their own objective functions. However, the objectives of all individual participants are not necessarily in line with system-wide objectives such as maximizing social welfare. Since the focus of this paper is on system-wide objectives, we review performance measures to be taken into account when generating ride-sharing matches. The *matching rate*, the *total vehicle-distance savings*, the *total travel time savings* and the *total finalization time* are the performance measures that we use to assess the quality of the solutions.

The total vehicle-distance savings represents the total distance driven by all participants traveling to their destinations, either in a shared ride or driving alone (Agatz et al., 2012). Similarly, the total travel time savings represents the total travel time spent by all the participants traveling to their destinations. Minimizing the total vehicle-distance and total travel time surrogate minimizing travel costs, that is a determinant factor for all the participants. Agatz et al. (2012) and Winter and Nittel (2006) used total vehicle-distance savings and total travel time savings, respectively, as performance measures.

The matching rate evaluates the number of finalized matches in the system. Unlike vehicle-distance savings which targets drivers and riders' benefits, improving the matching rate is desirable for all the stakeholders in the ride-sharing system, i.e. drivers, riders, and the ride-sharing provider. This criterion is all the more critical if the ride-sharing provider charges a commission per successful matches, either a fixed fee or a fare proportional to the trip cost. In addition, a higher matching rate may promote the use of the ride-sharing system, thus potentially leading to higher participants' utilities. This criterion is used by Winter and Nittel (2006), Ghoseiri et al. (2011), Amey (2011) and Xing et al. (2009) to evaluate their ride-sharing models.

These performance measures have frequently been used in the literature, individually or together, to evaluate model performance. In turn, measuring the finalization (trip notification) time of an accepted match has not receiving much attention. The trip finalization time can be used to assess customer experience since early notifications allow for better planning for both drivers and riders. In contrast, although delaying trip notification offers more flexibility from a system perspective, it does not guarantee that a better match will ultimately be found.

Since both the vehicle-distance savings and total travel time savings criteria are often highly correlated, we focus on one of these travel disutility criteria. In this paper, we consider the following performance measures to evaluate the performance of the proposed ride-sharing models:

1. Matching rate (MR): the total number of matched driver and rider announcements divided by the total number of trip announcements;
2. Average total vehicle-kilometres savings (AKS): total kilometres saved as a result of matching algorithms versus the scenario in which all individual trips are performed; and
3. Average finalization time (AFT): the average time between the announcement time and the time when the announcement is matched in minutes. A mathematical definition for the finalization time is provided in Section 6.2.

It should be highlighted that the above performance criteria may have different weights across different decision makers. For example, a small percentage improvement in the matching rate may have the same value for a decision maker as a higher relative improvement in the total distance savings.

## 3. Problem statement and formulation

In this section, we first present the mathematical formulation of the proposed ride-sharing system and introduce novel objective functions for the matching problem arising therein. In addition, we present a solution algorithm that combines a pre-processing procedure and a matching algorithm to solve the static ride-sharing problem.

### 3.1. Problem statement

Consider a set of drivers $D$ and a set of riders $R$. Let $a \in D \cup R$ be a trip announcement from a participant (driver or rider), we denote $\omega_a$ and $\delta_a$ the origin and the destination of this announcement, respectively. For each announcement $a$, we assume to know the announcement time $\tau(a)$, the earliest time at which the participant can depart from his/her origin $e(a)$, and its latest arrival time at his/her destination $l(a)$. In addition, the pairwise distances between all origin and destination locations $S(\omega_a,\delta_a)$ and the respective trip time $T(\omega_a,\delta_a)$ of each announcement $a$ are assumed to be known. The departure time window of announcement $a$ is then

$[e(a),q(a)]$ where $q(a) = l(a) - T(\omega_a,\delta_a)$ is the latest departure time of the participant. The length of the departure time window, $f(a) = q(a) - e(a)$, is hereby referred to as the *flexibility* of this participant.

Each announcement $a \in D \cup R$ is either a driver announcement ($D$) or a rider announcement ($R$). We henceforth denote $d$ a driver announcement and $r$ a rider announcement. As in Agatz et al. (2011), we assume that each shared ride consists of a single pick-up and a single drop-off. This assumption does not imply that multiple riders cannot be accommodated in a vehicle if they have the same origin and the same destination simultaneously. In this case, it is assumed that a single announcement will be considered. Hence, if a match between driver $d \in D$ and rider $r \in R$ is found, driver $d$ drives the distance between his/her origin and the origin of rider $r$, $S(\omega_d,\omega_r)$ to pick up the rider and then drives the distance $S(\omega_r,\delta_r)$ to the rider's destination $r_j$ before completing his/her trip by driving the distance $S(\delta_r,\delta_d)$ between the destination of the rider and his/her destination.

### 3.2. Pre-processing

In this section, we present a pre-processing procedure that aims to reduce the solution space by identifying infeasible matches and removing the corresponding driver-rider pair from consideration. A match between driver $d$ and rider $r$, henceforth referred to as the pair $(d,r)$, is said to be *feasible* if the time windows of driver $d$ and rider $r$ overlap such that there exists at least a pair of departure time and arriving time for the driver, and a pair of pick-up time and drop off time for the rider, that satisfy the time windows of both participants. To check for pair-feasibility we can calculate the overlap between the driver and the rider time windows by comparing the latest time that the driver $d$ must depart $k_d = \min[l(r) - T(\omega_r,\delta_r) - T(\omega_d,\omega_r), l(d) - T(\delta_r,\delta_d) - T(\omega_r,\delta_r) - T(\omega_d,\omega_r)]$ and the earliest departure times of both the driver and the rider. The match $(d,r)$ is feasible only if $\Delta T_d = k_d - Max[t,e(d)] \geqslant 0$ and if $\Delta T_r = k_d + T(\omega_d,\omega_r) - Max[t,e(r)] \geqslant 0$, where $t$ is the time at which the matching problem is solved.

We denote $P = \{(d,r): d \in D, r \in R\}$ the set of all pairs of drivers and riders. Before solving the ride-sharing matching problem, $P$ is pre-processed to identify the subset of feasible pairs according to the aforementioned feasibility check. Formally, let $\overline{P}$ be the set of feasible driver-rider pairs: $\overline{P} = \{(d,r) \in P: d \in D, r \in R, \Delta T_d \geqslant 0, \Delta T_r \geqslant 0,\}$. Limiting the set $P$ to $\overline{P}$ reduces the size of the matching problem by $|P| - |\overline{P}|$.

### 3.3. Matching problem formulation

Let $x_{dr}$ be a binary decision variable equal to 1 if the pair $(d,r)$ is matched, and 0 otherwise, and let $w_{dr} \geqslant 0$ be a weight representing the contribution of matching driver $d$ with rider $r$ in the objective function. The objective of the ride-sharing matching problem is to maximize the weighted sum $\sum_{(d,r)\in\overline{P}} w_{dr}x_{dr}$ and the complete formulation of the problem is summarized in Eqs. (1)–(4).

$$\max \sum_{(d,r)\in\overline{P}} w_{dr}x_{dr} \tag{1}$$

*Subject to:*

$$\sum_{r\in R:(d,r)\in\overline{P}} x_{dr} \leqslant 1 \quad \forall\, d \in D \tag{2}$$

$$\sum_{d\in D:(d,r)\in\overline{P}} x_{dr} \leqslant 1 \quad \forall\, r \in R \tag{3}$$

$$x_{dr} \in \{0,1\} \quad \forall\, (d,r) \in \overline{P} \tag{4}$$

According to Guillaume and Latapy (2006), the resulting mathematical problem is equivalent to a bipartite graph matching problem. Bipartite graphs are a particular class of graphs whose nodes can be divided into two disjoint sets, in which only the link between two nodes in different sets is permitted (Ou et al., 2007; Blattner et al., 2007). In the proposed ride-sharing model, driver-rider interactions can be represented as a bipartite graph $G$, such that each announcement is represented by a node and the nodes are classified into two sets of drivers ($D$) and riders ($R$).

Depending on whether the objective function of the matching problem uses uniform or non-uniform weights, a maximum cardinality matching algorithm or a maximum weight matching algorithm may be used for its resolution. For unweighted matching problems, the most popular approach is Hopcroft-Karp's algorithm (Hopcroft and Karp, 1973) whereas weighted matching problems are typically solved using the Hungarian algorithm (Harold, 1955), Ford-Fulkerson's algorithm (Ford and Fulkerson, 1956), Blossom algorithm (Edmonds, 1965a), or Edmonds-Karp's algorithm (Edmonds and Karp, 1972). A comprehensive review of matching algorithms can be found in Galil (1986).

In the matching problem, the weights $w_{dr}$ of each edge play a critical role in forming the best solution. Hence, these weights must be calculated in line with the ultimate objective of the ride-sharing system. We propose to evaluate the impact of using different weights in the matching problem objective function on the performance of the model. Specifically, we consider four weighting strategies discussed below.

**Maximizing the total net distance savings (DS):** A match results in distance savings only if the length of the matched trip – including the pick-up trip, the shared trip and the drop-off trip – $S_u(d,r) = S(\omega_d,\omega_r) + S(\omega_r,\delta_r) + S(\delta_r,\delta_d)$ is shorter than the sum of the lengths of the individual respective trips for both the driver and the rider $S_v(d,r) = S(\omega_d,\delta_d) + S(\omega_r,\delta_r)$. Therefore, the net distance savings is $\Delta S(d,r) = S_v(d,r) - S_u(d,r)$. The motivation behind using net distance savings in the objective function is to reduce travel cost, which in turn increases participants' utilities. Formally, this is achieved by replacing $w_{dr}$ in Eq. (1) with $\Delta S(d,r)$ for each pair

$(d,r) \in \overline{P}$ and gives the objective function:

$$\max \sum_{(d,r)\in\overline{P}} \Delta S(d,r) x_{dr} \tag{5}$$

**Maximizing the total number of matches (NM):** The number of matched announcements is critical for the long-term sustainability of a ride-sharing service to ensure customer satisfaction and profitability (Stiglic et al., 2016). Hence, maximizing the number of matches can be interpreted as a measure of reliability of the ride-sharing system (Nourinejad and Roorda, 2016). Formally, this objective is modelled with uniform weights $w_{dr} = 1$ in the objective function:

$$\max \sum_{(d,r)\in\overline{P}} x_{dr} \tag{6}$$

**Maximizing the total distance proximity index (DP):** We propose a new index that can be used as the weight in the ride-sharing matching problem based on the proximity of the driver and the rider initial trips, hereby referred to as the DP index. The DP index is defined in Eq. (7).

$$DP(d,r) = \min\left(\frac{S(\omega_d,\delta_d)}{S(\omega_r,\delta_r)},\frac{S(\omega_r,\delta_r)}{S(\omega_d,\delta_d)}\right) \tag{7}$$

DP is conceptually different from NM and DS so that this objective function is not directly related to the performance measures of the system, i.e. MR and AKS. DP indirectly steers the optimization in the right direction within the rolling horizon framework.

The intuition behing the DP index is that driver and rider trips of similar distance will be good match if their origin and destinations are in close vicinity. In particular, this applies to commuting trips. In turn, it is likely that trips with similar distances have spatially correlated origins and destinations. While it is worth noting that commute distance is blind to the direction of the trip, extensive computational experiments suggest that the DP index is useful in revealing information about potential trips with similar properties, especially if trips are temporally correlated. The effectiveness of this hypothesis is examined later in the paper (see Section 6.2). Using the DP index in the objective function, we get:

$$\max \sum_{(d,r)\in\overline{P}} DP(d,r) x_{dr} \tag{8}$$

The DP index does not take into account the spatial correlation between origins and destinations or trips. To illustrate this limitation, consider the problem depicted in Fig. 1 which shows two pairs with identical DP indices. In the first pair, the rider's trip is twice longer than the driver's while in the second pair, the driver's trip length is twice longer than rider's. In both cases the DP index is 0.5, however, the length of the matched trip for the second pair is 3 compared to 8 for the first pair. This highlights the need for considering the length of the matched trip, i.e. from the driver's origin to the rider's origin, from the rider's origin to the rider's destination, and from the rider's destination to the driver's destination, within the objective function. Therefore, we propose an adjustment factor to the DP index to accommodate this.

**Maximizing the total adjusted distance proximity index (ADP):** To include the impact of the length of the matched trip into the total trip length, we define an adjustment factor (AF) for pair $(d,r)$ as follows:

$$AF(d,r) = \frac{S(\omega_d,\delta_d)}{S(\omega_d,\omega_r) + S(\omega_r,\delta_r) + S(\delta_r,\delta_d)} \tag{9}$$

$AF(d,r)$ measures the length of driver's individual compared to the length of the matched trip. Matches with smaller matched trip distances, in comparison with the driver initial trip, should receive higher priority in the objective function. Hence, we introduce the ADP index as defined in Eq. (10).

$$ADP(d,r) = AF(d,r) \times DP(d,r) \tag{10}$$

Using the ADP index in the objective function, we get:

$$\max \sum_{(d,r)\in\overline{P}} ADP(d,r) x_{dr} \tag{11}$$

In Section 6.2, we will show that both DP and ADP indices lead to non-dominated solutions.

To illustrate the influence of $w_{dr}$ in the matching problem, consider the ride-sharing problem depicted in Fig. 2. In this example,

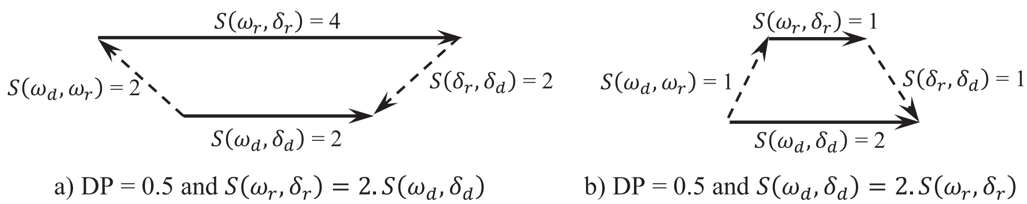



**Fig. 1.** Different matched trip distances for identical DP indices.

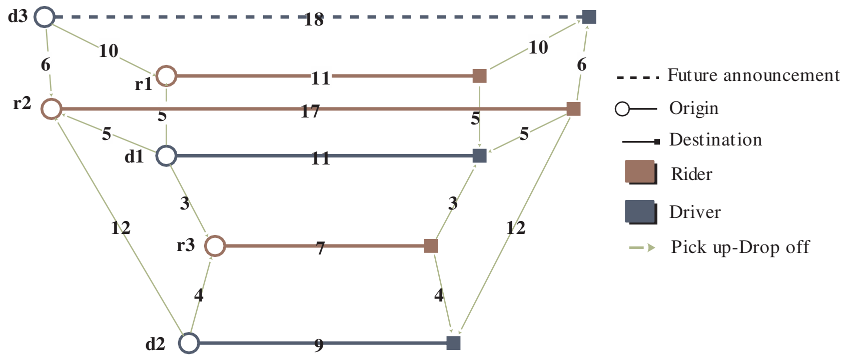



**Fig. 2.** A ride-sharing problem.

there are two drivers ($d1$,$d2$) and three riders ($r1$,$r2$,$r3$) in the ride-sharing system at time period $t$. Assume that the pairs ($d2$,$r1$) and ($d2$,$r2$) cannot be matched due to incompatible time windows. The possible distance savings are: $\Delta S(d1,r1) = 1$,$\Delta S(d1,r2) = 1$,$\Delta S(d1,r3) = 5$, and $\Delta S(d2,r3) = 1$. If $DS$ is used as the objective function (i.e. $w_{dr} = \Delta S(d,r)$), solving the matching problem results in $x_{d1,r3} = 1$ while if $NM$ is used (i.e. $w_{dr} = 1$) the optimization problem results in either $x_{d1,r1} = 1$ and $x_{d2,r3} = 1$; or $x_{d1,r2} = 1$ and $x_{d2,r3} = 1$. In turn, if $DP$ is used the following values are obtained: $DP(d1,r1) = 1$, $DP(d1,r2) = \frac{11}{17}$, $DP(d1,r3) = \frac{7}{11}$ and $DP(d2,r3) = \frac{7}{9}$. Therefore, if $DP$ is used as the objective function, we obtain $x_{d1,r1} = 1$ and $x_{d2,r3} = 1$. Finally, using ADP as the objective function we get: $ADP(d1,r1) = \frac{11}{21}$, $ADP(d1,r2) = \frac{11}{17} \times \frac{11}{27}$, $ADP(d1,r3) = \frac{7}{11} \times \frac{11}{13}$ and $ADP(d2,r3) = \frac{7}{9} \times \frac{9}{15}$ which results in $x_{d1,r3} = 1$ and $x_{d2,r3} = 1$.

Assume now that a new driver $d3$ enters the system at $t + 1$. Based on the outcome of the matching problem at time period $t$, $d3$ can be either matched to $r1$ or $r2$; or remain unmatched if both riders have already been matched.

### 3.4. DS ε -condition

If DS is used in the objective function a matched pair $(d,r)$ will verify the condition such that: $\Delta S(d,r) \geqslant 0$. In turn, this may not be the case if DP, ADP or NM is used in the objective function. Instead the corresponding solutions may lead big negative distance savings, which may considerably compromise the overall quality of the solution. To balance the performance of the ride-sharing system across the performance measures (MR, AKS), we propose to use the condition $\Delta S(d,r) \geqslant \varepsilon$ in the pre-processing procedure to exclude pairs with distance savings lower than ε. For this purpose, we propose to extend the formulation (1)–(4) by adding side-constraints of the form:

$$\Delta S(d,r) \geqslant \varepsilon \tag{i}$$

This condition aims to help in combining multiple criteria in the optimization.

### 3.5. Static solution algorithm

The steps that are illustrated in the previous sub-sections are summarized in Algorithm 1, henceforth referred to as STATIC. This algorithm will be used at each iteration of the dynamic ride-sharing algorithms to periodically solve the matching problem as new announcements enter the system.

**Algorithm 1** (*Static*).

**Input:** Set of driver announcements $D$, set of rider announcements $R$, Objective function (DS, NM, DP or ADP)
**Output:** A weighted bipartite graph $G$, a matching vector $\boldsymbol{x}$ for all pairs of driver-rider in $\overline{P}$
$P \leftarrow \{(d,r): d \in D, r \in R\}$
$\overline{P} \leftarrow \varnothing$
**for** $(d,r)$ in $P$:
  **if** $\Delta T_d \geqslant 0$, $\Delta T_r \geqslant 0$, and $\Delta S(d,r) \geqslant \varepsilon$ **then**:
    $\overline{P} \leftarrow \overline{P} \cup \{(d,r)\}$
  **end if**
**end for**
$\boldsymbol{w} \leftarrow$ Determine weight vector based on the objective function and $\overline{P}$
$G \leftarrow (D \cup R, \overline{P}, \boldsymbol{w})$
$\boldsymbol{x} \leftarrow$ Execute the maximum-weight bipartite matching algorithm on $G$

## 4. Rolling horizon framework

Driver and rider announcements may enter the ride-sharing system continuously at any time, thus making the problem dynamic and calling for event-driven modelling frameworks. In this section, we present a *rolling horizon* approach to solve the on-demand ride-sharing problem in real-time. The rolling horizon approach is motivated by the following. First, the system is assumed to be highly dynamic so that driver and rider requests enter and leave the system continuously. Second, due to the unpredictable nature of the problem, future driver and rider requests are assumed to be unknown.

There exist several approaches to determine the frequency of the iterations in a rolling horizon framework. The main two approaches include periodic optimization with fixed time step and event-driven optimization where an event can include a new announcement or a batch of new announcements. However, event-driven approaches are usually used when systems are expected to react quickly to changes in their environment while periodical optimization may imply longer reaction delay (Pillac et al., 2012). Moreover, defining proper "events" requires sophisticated techniques compared to simply reacting to time steps that is the cost of optimizing very complicated systems. The appropriate approach is highly dependent on the policy and the time step that is used to finalize matches. In this paper, we adopt the former approach and assume that the rolling horizon algorithm is executed for a given set $T$ of time steps, i.e.: $T = \{0,p,2p,3p,...\}$.

In the proposed ride-sharing system, we adopt an infinite look-ahead time horizon approach to slide the horizon with a predefined time step $p$, and then execute the STATIC algorithm to update the pairs periodically (for each time step). Then, at each iteration of the rolling horizon, the matching problem presented in Section 3.3 is solved with one of the proposed objective functions (DS, NM, DP or ADP) for the set of active announcements. At any time $t \in T$, an announcement is labelled *active* if its latest departure time is greater than the execution time of the STATIC algorithm, i.e. $q(a) \geqslant t$. Further, an announcement is labelled *inactive* if it is matched to another announcement or *expired* if $q(a) < t$. The proposed rolling horizon approach may allow planners to handle demand uncertainty since decisions made based on the current available data may be re-considered at a later stage as long as the associated announcements have not expired.

At each iteration of the rolling horizon framework, a matched pair (d,r) can either be finalized, i.e. the match is accepted by the ride-sharing system, or its finalization can be delayed. Postponing the finalization time can be advantageous if a better match for either the driver, the rider or both of them can be found in the future. We henceforth refer to this process as the *dynamic matching policy*. Further, we denote $\overline{x}_{dr}$ the finalized value of variable $x_{dr}$, i.e. $\overline{x}_{dr} = 1$ means that the match $(d,r)$ is accepted by the ride-sharing system and $\overline{x}_{dr} = 0$ means that at least $d$ or $r$ is involved in another finalized match, thus these two announcements will not be matched together. If a match is finalized, the respective driver and rider exit the system and will not be considered at the next iteration. This rolling horizon iterates until all announcement exit the system either by being matched or by having expired.

The rolling horizon approach cannot ensure the goodness of the solution (Li and Ierapetritou, 2010) since, at each period of time, some of the matches are finalized and cannot be reconsidered. Thus, it should be supported with a high-quality optimization procedure. In a real-time ride-sharing system, choosing a suitable objective function and matching policy, can significantly impact the quality of the solutions.

We next present three classes of dynamic matching policies that can be used within the proposed ride-sharing system to decide when driver-rider matches and the corresponding trip should be finalized.

### 4.1. As late as possible (ALAP) dynamic matching policy

From a system perspective, the most versatile policy consists in finalizing trips *as late as possible*: under this policy, matched trips are not finalized until the next time period exceeds the latest departure time of either driver or rider, $q(d) < t + p$ or $q(r) < t + p$. Therefore, a match is finalized at time period $t \in T$ if the *finalization condition* $x_{dr} = 1$ and $\min[q(d),q(r)] < t + p$ is satisfied.

### 4.2. As soon as possible (ASAP) dynamic matching policy

An alternative dynamic matching policy consists in finalizing matches *as soon as possible*: under this policy, any matched trip among drivers and riders is finalized on its first occurrence within the rolling horizon algorithm. The finalization condition for the ASAP policy is $x_{dr} = 1$. This policy provides less flexibility from a system perspective but potentially offers a better service to users in that they are likely to be notified significantly earlier than under the ALAP policy.

### 4.3. As soon as α (ASA α) dynamic matching policy

The third policy discussed in this paper finalizes matches as soon as a condition $\alpha$ is met. The idea behind this approach is to borrow advantages from both the ASAP and ALAP policies. In the ASA $\alpha$ policy, a match is finalized if the next time period exceeds the latest departure time of either the driver or the rider; or if the weight of the matched announcements in the objective function exceeds the critical value of $\alpha$. The finalization condition for $ASA\alpha$ is $x_{dr} = 1$ and either $\min[q(d),q(r)] < t + p$ or $w_{dr} \geqslant \alpha$. This policy collapses to the ASAP policy if $\alpha = 0$.

The pseudo code of the dynamic matching policies algorithm is presented in Algorithm 2 and henceforth referred to as the ROLLING HORIZON algorithm.

**Algorithm 2** (*ROLLING HORIZON*).

```
Input: Announcement s sets D and R, objective function (DS, NM, DP, ADP), dynamic matching policy (policy)
Output: A finalized vector x̄ = [x̄_dr]
for t ∈ {0,p,2p,3p,...}:
  D_t ← {a ∈ D: q(a) ⩾ t,τ(a) ⩽ t}
  R_t ← {a ∈ R: q(a) ⩾ t,τ(a) ⩽ t}
  G ← STATIC (D_t, R_t, objective function)
  x ← Execute the maximum-weight bipartite matching algorithm on G
  for (d,r) ∈ P̄_t: x_dr = 1:
    if policy = ALAP then:
       if min[q(d),q(r)] < t + p:
          x̄_dr ← 1
          D_t ← D_t \ {d}
          R_t ← R_t \ {r}
       end if
     else if policy = ASAP then:
       x̄_dr ← 1
       D_t ← D_t \ {d}
       R_t ← R_t \ {r}
     else if policy = ASA α then:
       if (min[q(d),q(r)] < t + p) or w_dr ⩾ α
          x̄_dr ← 1
          D_t ← D_t \ {d}
          R_t ← R_t \ {r}
       end if
     end if
   end for
   for a_i,j ∈(D_t ⋃ R_t):
     if q(a) < t + p then
       remove the announcement from either D_t or R_t
     end if
   end for
end for
```

To illustrate the impact of dynamic matching policies, Fig. 3 depicts an example of a ride-sharing system with two drivers and two riders. In this example, the time window of $r1$ overlaps with that of both $d1$ and $d2$ while the time window of $r2$ only overlaps with that of $d2$. Assume that the feasible matches set is $\overline{P} = \{(d1,r1),(d2,r1),(d2,r2)\}$. For each announcement, a match can be found only in the time window between the time when announcement $a$ is received $\tau(a)$, and the time step prior to its latest departure time $q(a)$. If the Policy ASAP is used, the match $(d1,r1)$ will be finalized at $t = p$. The setting of the system can be changed so that the decision on finalization of the found match to be made whenever in between $\tau(a)$ and $q(a)$. In the example, the system may postpone the assignment of $r1$ until its latest departure time $q(r1)$ in the hope to find a better match at a later time period. For instance, the system can find the pair $(d2,r1)$ to be a better match for $r1$ at $t = 2\ p$.

This example and the example provided in Section 3.3 highlight that the ride-sharing problem is highly dynamic and that the

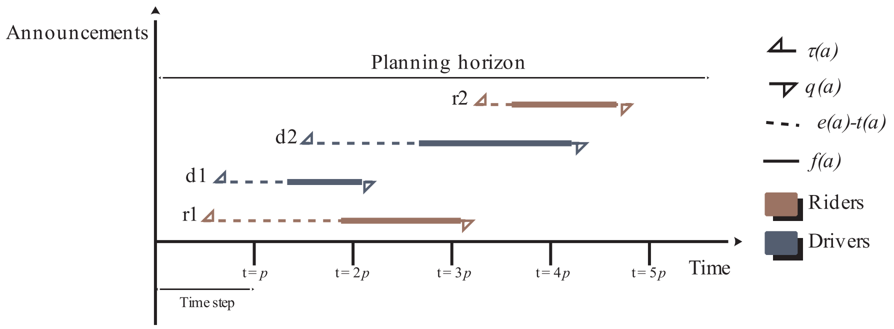



**Fig. 3.** Dynamic policies scheme.

selected objective function and matching policy can considerably impact its solution.

## 5. Clustering heuristic

Although the pre-processing steps and the maximum weighted bipartite matching algorithms have polynomial-time worst case time complexities, solving instances with a large number of participants in real-time can be challenging. Given the dynamic component of the problem, the algorithms must be solved periodically over short periods of time, thus emphasizing the need for large-scale solution methods. In this section, we propose a novel clustering algorithm based on $k$-means clustering (Lloyd, 1982) to address this challenge.

The proposed clustering algorithm attempts to solve the original ride-sharing problem by assigning the active announcements to a number of smaller sub-problems that are faster to solve. We first present the clustering algorithm in a static context, where all the announcements are assumed to be available and then explain how this algorithm can be embedded within the proposed ROLLING HORIZON algorithm.

Let $|N|$ be the desired number of clusters and let $N$ be the set of such clusters. We assume that the ride-sharing announcements $a \in D \cup R$ can be divided into potentially intersecting subsets $D_n$ and $R_n$ for $n \in N$ where $D_n$ and $R_n$ are the sets of driver and rider announcements assigned to cluster $n$. We first use the latitude and the longitude of the origin $\omega_a$ and destination $\delta_a$ of driver announcements $a \in D$ to create $|N|$ clusters using the $k$-means algorithm. Then, the origin and destination of riders are assigned to the closest cluster based on their spatial coordinates. Here closeness is defined as the Euclidean distance from the rider's origin or destination to the centroid of the cluster (Lloyd, 1982).

At this point the origin and destination of an announcement (driver or rider) may not belong to the same cluster. In this case, the corresponding announcement is assigned to both clusters; otherwise, both the origin and destination belong to the same cluster and the announcement is assigned to this cluster only (see Fig. 4). The pseudo-code of the clustering algorithm is presented in Algorithm 3.

**Algorithm 3** (*CLUSTERING*).

**Input:** number of clusters $|N|$, $D$, $R$
**Output:** $\{D_n, R_n : n \in N\}$
$D_n \leftarrow \varnothing$
$R_n \leftarrow \varnothing$
$X \leftarrow \{\omega_a, \delta_a : a \in D, q(a) \geqslant t, \tau(a) \leqslant t\}$
$\{D_n : n \in N\} \leftarrow$ k-means($|N|$, $X$)
**for** $a$ in $R$:

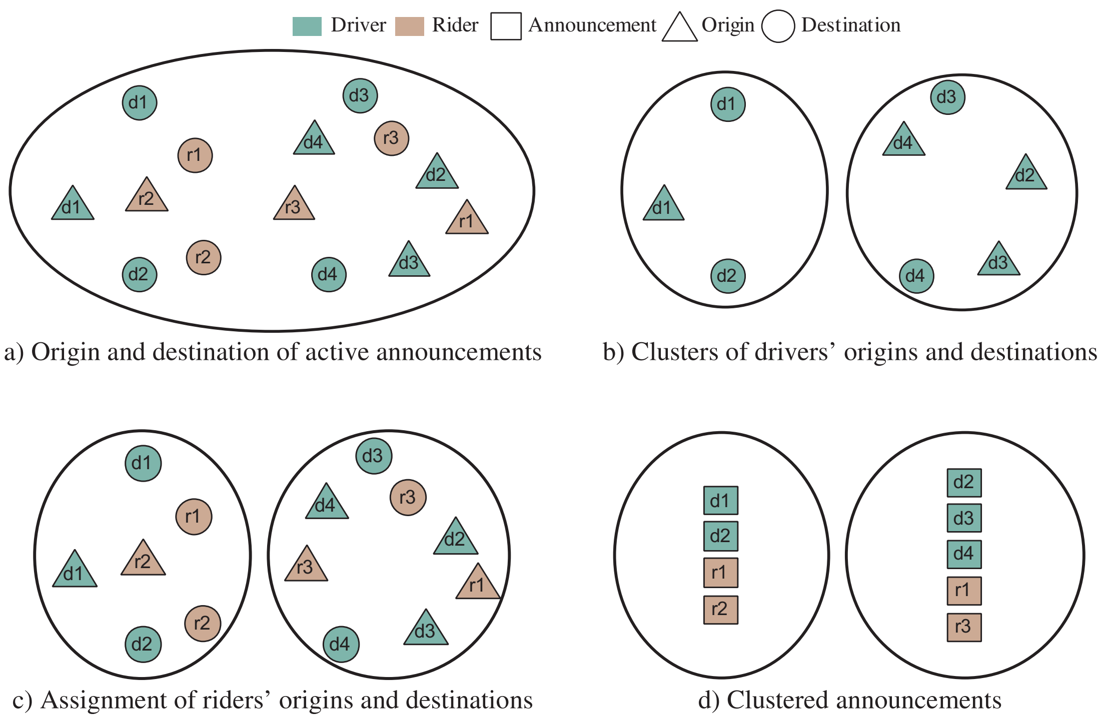



**Fig. 4.** Illustration of the intersecting clustering algorithm with $|N| = 2$.

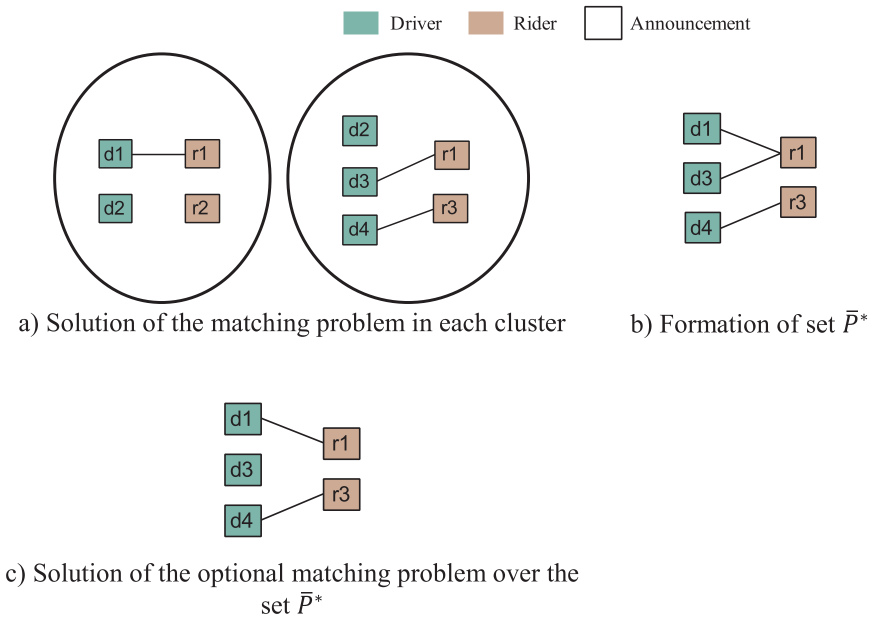



**Fig. 5.** Cluster-based matching with $|N| = 2$.

  $n\leftarrow$ find the closest cluster to $\omega_a$
  $R_n \leftarrow R_n \cup \{a\}$
  $n\leftarrow$ find the closest cluster to $\delta_a$
  $R_n \leftarrow R_n \cup \{a\}$
**end for**

For each cluster $n \in N$, the matching problem is then solved and the resulting matched pairs are stored in a set $\bar{P}_n^*$. Should an announcement be matched twice (i.e. this announcement is matched in two clusters), the resulting solution is not feasible and we solve another matching problem among the set of matched pairs $\bar{P}^* = \bigcup_{n\in N}\bar{P}_n^*$. This optional matching step ensures that the final solution is feasible, i.e. an announcement can be matched at most once (see Fig. 5). The pseudo-code of the cluster-based matching step is summarized in Algorithm 4.

To embed the clustering heuristic within an on-demand ride-sharing framework, we propose to periodically re-cluster the announcements and use the proposed cluster-based approach to solve the matching problems therein. In particular, to save computation time, the clustering heuristic may be executed less frequently than the matching problems are solved. Extensive testing suggests that this strategy provides a good compromise between computational runtime and solution quality. Specifically, we introduce a set $T'$ of time steps, i.e.: $T' = \{p',2p',3p',...\}$ where $p' \geqslant p$. At each iteration $t$ of the ROLLING HORIZON algorithm, we check if $t \geqslant kp'$ where $k$ is an integer incremented after each execution of the clustering heuristic and re-cluster the announcement if this condition is verified. Hence, clusters are only recalculated with a period $p'$ following the CLUSTERING procedure, as depicted in Fig. 4. In turn, the matching problems are solved based on the cluster-based approach as depicted in Fig. 5, i.e. new announcements entering the system at each time period $t \in \{p,2p,3p,...\}$ are assigned to the current clusters, a matching problem is solved for each cluster and an optional matching problem is solved if one or more announcements are assigned and matched in more than one cluster.

**Algorithm 4** (*Cluster-based Matching*).

**Input:** $\{D_n,R_n: n \in N\}$
**Output:** a matching vector $\boldsymbol{x}$
$D^* \leftarrow \varnothing$
$R^* \leftarrow \varnothing$
**for** $n \in N$:
  $\bar{P}_n^*\leftarrow$ STATIC ($D_n$, $R_n$, objective function)
**end for**
**if** an announcement has been matched twice **then**:
  $\bar{P}^* = \bigcup_{n\in N}\bar{P}_n^*$
  $G\leftarrow$ form the graph based on the matched pairs in $\bar{P}^*$
  $\boldsymbol{x}\leftarrow$ Execute the maximum-weight bipartite matching algorithm on $G$

**else**:
$\boldsymbol{x} \leftarrow$ Construct matching vector using all sets $\overline{P}_n^*$ for $n \in N$

To evaluate the impact of the number of clusters onto solution quality and computational performance, we conduct a sensitivity analysis of this input in Section 6.3.

## 6. Numerical experiments

In this section, we evaluate the performance of the proposed objective functions and dynamic matching policies introduced in Section 3 and Section 4 as well as the proposed clustering algorithm in Section 5.

### 6.1. Data and simulation

To test and validate the proposed ride-sharing system, we use data from the Melbourne metropolitan area, Australia. Melbourne is the second most populated city of Australia and capital of the state of Victoria with the population of 4.88 million spread across an area of about 10,000 km$^2$. We used Zenith strategic transport/land use model provided by Veitch Lister Consulting to build the simulation scenarios (VLC, 2013). Zenith is a large-scale, multi-modal travel model which is implemented in OmniTRANS software package, and currently operates in eight Australian cities and works by simulating the daily travel behaviour of all the residents (and visitors). Zenith uses road and public transport networks, tolls, parking charges, public transport fares and vehicle operating costs, demographic, land use data of 4003 geographic zones in Victoria. The model outputs used in the simulation exercise of this paper are travel volumes, travel time, and origin-destination demand volume and origin-destination distance matrices.

As a result, three demand scenarios of 0.25%, 0.5% and 0.75% participation rates are considered between 6:00am and 9:00 pm. In these scenarios, driver and rider announcements are randomly generated based on time of day and the inter-zonal density of trips. Drivers are randomly chosen from trips that are made by single occupant cars; while riders are randomly chosen from transit based and multiple occupancy trips. According to the Integrated Survey of Travel and Activities (VISTA) in 2007 (Department of Transport, 2009), the total number of trips in Melbourne is about 13.5 million, with 55% by private cars and around 30% by public transport or as a passenger. According to Parker (2004), 70% of the vehicles in Melbourne are single occupancy vehicles. As a result, we assume that the percentage of trips by single occupancy cars is 38.5%. Ten random streams of trips per participation rate are generated to be able to accurately compare and assess the performance of different combinations of objective functions and dynamic matching policies.

Although departure time and travel time of each trip is available, the above dataset does not provide information on the participants' travel time window. As in Agatz et al. (2011), we fixed trip flexibility rather allowing it to be a function of travel time. This prevents underestimating or overestimating the flexibility of short and long trips, respectively. Further, we did not differentiate between different trip purposes. Let $b$ and $c$ be departure time and travel time of a trip generated. We determine its earliest departure time $e$ and its latest arrival time $l$ as $e = b - 10$ and $l = b + c + 10$ (in minutes), respectively. Finally, we randomly generate announcement times using a uniform distribution with parameters ($b$ - 60, $b$).

In all experiments, a period of $p = 2$ min (time step) is used for the rolling horizon algorithm and a total of 450 time periods are considered ($|T| = 450$). Further, a period of $p' = 20$ minutes is used in the proposed clustering algorithm. We assume full compliance of the participants, i.e. if a ride is finalized, then the corresponding driver and rider are immediately notified and accept the trip.

To illustrate the scale of the instances considered, the stream with the participation rate of 0.5% corresponds to 25,987 drivers and 20,250 riders. In one of the streams picked randomly among the generated random streams with 0.75% participation rate, on average 154 announcements (69,355/450) and in the worst case, 313 new announcements enter the system in each time step. Nonetheless, the number of active announcements (accumulated entered announcements at previous time periods that are still active) may be much larger than this value. For example, in the noted randomly picked sample, for the ALAP policy, on average 971 announcements and, in the worst case, 1910 announcements are active in the system.

All algorithms of the proposed ride-sharing system are implemented in Python 3.5 on a machine with 16 Gb of RAM with a processor of i7-4770. We use Hopcroft-Karp's algorithm (Hopcroft and Karp, 1973) and Edmonds' algorithm (Edmonds 1965a, 1965b) to solve the unweighted (for the NM) and weighted (for the DP, ADP and DS) matching problems which are implemented in Networkx, a Python package for scientific computing applications which includes a suite of network algorithms implemented based on seminal research papers (Hagberg et al., 2008). In the Networkx implementation, Hopcroft-Karp's and Edmonds' algorithms have a worst-case time complexity of $O(|E|\sqrt{|V|})$ and $O(|V|^3)$, respectively; where $E$ and $V$ are the set of edges and the set of nodes in the network.

### 6.2. Computational results

In this section, we present the simulation results of the scenarios explained before. For each experiment, we evaluate the performance of objective functions and the dynamic matching policies presented in Sections 3.3 and 4. Then, we compare the quality of their solutions with regard to the performance measures discussed in Section 2.3. For the AFT, the finalization time for each finalized match is calculated as the sum of the waiting times of the driver, $t - \tau(d)$, and that of the rider, $t - \tau(r)$.

**Table 1**
Objective functions' performance measures in the static model.

| Objective function | MR (%) | AKS (%) |
|---|---|---|
| DS | 15.47 | 4.21 |
| DP | 58.66 | −7.66 |
| ADP | 58.67 | −6.76 |
| NM | 58.68 | −7.23 |

### 6.2.1. Static problem benchmark

In a static ride-sharing problem wherein all announcements are known prior to the start of the day, performance is assessed. The results for participation rate of 0.25% are provided in Table 1: for this static case, we focus on the MR and AKS. We find that the performance of objectives DP and ADP are very close to the optimal MR (58.68%) obtained using the NM objective. This suggests that the DP and ADP indices work toward maximizing the number of matches.

In turn, the results using DS are quite different from that of the other objectives: this objective yields a much lower MR but improves on the AKS achieved by the other objectives and provides a positive (4.21%) AKS.

Fig. 6 depicts the performance of the ride-sharing system with respect to MR and AKS for the static benchmark with different pairwise distance saving thresholds, i.e.: $\varepsilon = \{-10,-9,\ldots,0,\ldots,10\}$. The figure depicts similar performance for DP, ADP and NM with regard to MR across different $\varepsilon$ values; whereas, there is a noticeable difference among the objective functions with regard to the AKS. This shows that proposed indices tend to significantly improve the AKS while maximizing the MR. Further, the AKS performance of DP and ADP objective functions approach the values obtained from optimizing the DS objective function. The figure shows that the proposed objective functions DP and ADP can find non-dominated solutions in terms of the AKS and MR performance measures.

### 6.2.2. Dynamic problem benchmark

To conduct the dynamic problem analysis, the values of 0 and $-5$ are chosen for $\varepsilon$ in the condition $\Delta S(d,r) \geqslant \varepsilon$ for filtering matches in the DS $\varepsilon$-condition setting. The first value guarantees cost reduction for each matched drivers and riders (assuming that the travel costs are proportional to the travelled distances). The second value for $\varepsilon$ is chosen based on the static benchmark conducted in Section 6.2.1. Specifically, for this setting the performance measure AKS remains near zero, thus providing a more flexible matching condition while ensuring that the overall system performance beneficial from a travel cost perspective.

To evaluate different solutions, especially solutions generated by the proposed DP and ADP objective functions, NM-Static (in this paper, for instance, NM-Static refers to a NM objective function that is implemented in a static problem) and DS-Static are used to obtain extreme values for NM and AKS, respectively. Table 2 and Fig. 7 summarize the results of multiple combinations of objective functions and dynamic matching policies. Specifically, we consider different combinations of NM, DP, ADP and DS objective functions and dynamic matching policies of ASAP, ASA $\alpha$ and ALAP. Further, the static versions of the model for each of the objective functions, as the benchmarks, are reported in the table. Since negative weights are always ignored in the maximum weight matching problem, using DS as the objective function produces similar results with both $\varepsilon = -5$ and $\varepsilon = 0$, hence only one set of results is presented for this objective function.

Comparing ASAP and ALAP policies with the static approach highlights the following issues: First, the gap between the static values and the ASAP policy measures demonstrates the potential improvement when prior information becomes available through participants before finalizing matches. When no prior information is received, a better solution might be obtained if finalization of matches is postponed using a rolling horizon based policy. Further, although the ALAP policy may improve the quality of solutions in terms of MR and AKS, it significantly increases the AFT value. Hence, from a customer service perspective, there exists a trade-off

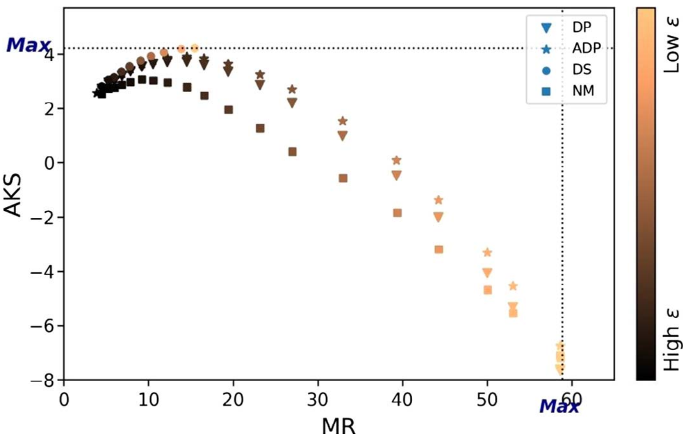



**Fig. 6.** Performance of different objective functions on the static ride-sharing problem (participation rate = 0.25%).

**Table 2**
Performance measures for the dynamic problem benchmark.

| Objective function | Dynamic matching policy | Participation rate | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 0.25% | | | 0.5% | | | 0.75% | | |
| | | MR (%) | AKS (%) | AFT | MR (%) | AKS (%) | AFT | MR (%) | AKS (%) | AFT |
| *NM* ($\varepsilon = 0$) | ASAP | 13.44 | 2.26 | 7.55 | 16.07 | 2.66 | 6.68 | 17.05 | 2.86 | 6.21 |
| | ALAP | 14.26 | 2.27 | 26.17 | 17.01 | 2.75 | 25.21 | 17.97 | 2.91 | 25.83 |
| | Static | 16.61 | 2.47 | – | 19.12 | 2.91 | – | 19.96 | 3.02 | – |
| *NM* ($\varepsilon = -5$) | ASAP | 31.71 | −1.35 | 6.41 | 35.50 | −1.37 | 5.65 | 36.76 | −1.31 | 5.16 |
| | ALAP | 34.79 | −1.50 | 25.95 | 38.73 | −1.55 | 25.16 | 39.99 | −1.38 | 25.58 |
| | Static | 39.40 | −1.85 | – | 42.74 | −1.75 | – | 43.68 | −1.90 | – |
| *DP* ($\varepsilon = 0$) | ASAP | 13.45 | 2.52 | 7.61 | 16.08 | 3.27 | 6.76 | 17.14 | 3.63 | 6.19 |
| | ASA $\alpha = 0.1$ | 13.49 | 2.58 | 7.68 | 16.09 | 3.27 | 6.88 | 17.15 | 3.65 | 6.43 |
| | ASA $\alpha = 0.2$ | 13.54 | 2.69 | 8.56 | 16.14 | 3.44 | 7.49 | 17.16 | 3.82 | 7.00 |
| | ASA $\alpha = 0.3$ | 13.67 | 2.81 | 10.38 | 16.19 | 3.57 | 8.79 | 17.20 | 3.98 | 8.13 |
| | ASA $\alpha = 0.4$ | 13.81 | 2.88 | 13.41 | 16.34 | 3.67 | 11.30 | 17.36 | 4.08 | 10.40 |
| | ASA $\alpha = 0.5$ | 13.91 | 2.90 | 15.52 | 16.47 | 3.71 | 13.17 | 17.47 | 4.13 | 12.18 |
| | ASA $\alpha = 0.6$ | 14.02 | 2.93 | 17.56 | 16.61 | 3.75 | 15.19 | 17.64 | 4.16 | 14.10 |
| | ASA $\alpha = 0.7$ | 14.06 | 2.92 | 19.08 | 16.74 | 3.76 | 16.81 | 17.73 | 4.17 | 15.90 |
| | ASA $\alpha = 0.8$ | 14.10 | 2.93 | 20.38 | 16.81 | 3.76 | 18.22 | 17.81 | 4.18 | 17.28 |
| | ASA $\alpha = 0.9$ | 14.14 | 2.93 | 21.33 | 16.88 | 3.77 | 19.44 | 17.85 | 4.17 | 18.55 |
| | ALAP | 14.27 | 2.96 | 26.08 | 17.00 | 3.80 | 25.67 | 17.99 | 4.21 | 25.68 |
| | Static | 16.57 | 3.60 | – | 19.08 | 4.49 | – | 19.92 | 4.89 | – |
| *DP* ($\varepsilon = -5$) | ASAP | 32.54 | −1.04 | 6.63 | 36.24 | −0.60 | 5.90 | 37.54 | −0.35 | 5.33 |
| | ASA $\alpha = 0.1$ | 32.59 | −0.90 | 6.61 | 36.42 | −0.64 | 5.88 | 37.77 | −0.33 | 5.46 |
| | ASA $\alpha = 0.2$ | 32.77 | −0.79 | 7.13 | 36.53 | −0.43 | 6.24 | 37.83 | −0.15 | 5.77 |
| | ASA $\alpha = 0.3$ | 33.06 | −0.61 | 8.33 | 36.72 | −0.20 | 7.12 | 37.98 | 0.10 | 6.53 |
| | ASA $\alpha = 0.4$ | 33.40 | −0.53 | 10.07 | 37.09 | −0.09 | 8.44 | 38.28 | 0.25 | 7.76 |
| | ASA $\alpha = 0.5$ | 33.83 | −0.50 | 12.20 | 37.48 | −0.03 | 10.19 | 38.63 | 0.30 | 9.34 |
| | ASA $\alpha = 0.6$ | 34.19 | −0.56 | 13.98 | 37.89 | −0.06 | 11.85 | 38.98 | 0.27 | 10.79 |
| | ASA $\alpha = 0.7$ | 34.44 | −0.60 | 15.63 | 38.18 | −0.11 | 13.40 | 39.31 | 0.21 | 12.36 |
| | ASA $\alpha = 0.8$ | 34.59 | −0.61 | 16.87 | 38.37 | −0.16 | 14.69 | 39.53 | 0.18 | 13.67 |
| | ASA $\alpha = 0.9$ | 34.67 | −0.62 | 17.73 | 38.49 | −0.16 | 15.63 | 39.64 | 0.16 | 14.57 |
| | ALAP | 34.89 | −0.66 | 26.04 | 38.78 | −0.21 | 25.93 | 40.00 | 0.11 | 25.77 |
| | Static | 39.30 | −0.48 | – | 42.65 | 0.11 | – | 43.60 | 0.43 | – |
| *ADP* ($\varepsilon = 0$) | ASAP | 13.49 | 2.67 | 7.57 | 16.13 | 3.29 | 6.77 | 17.12 | 3.69 | 6.30 |
| | ASA $\alpha = 0.1$ | 13.39 | 2.72 | 7.95 | 16.00 | 3.35 | 6.86 | 17.11 | 3.73 | 6.44 |
| | ASA $\alpha = 0.20$ | 13.39 | 2.79 | 9.20 | 16.02 | 3.57 | 7.76 | 17.12 | 3.94 | 7.16 |
| | ASA $\alpha = 0.25$ | 13.47 | 2.83 | 10.62 | 16.06 | 3.67 | 8.91 | 17.15 | 4.06 | 8.09 |
| | ASA $\alpha = 0.30$ | 13.54 | 2.91 | 12.17 | 16.23 | 3.78 | 10.20 | 17.25 | 4.17 | 9.44 |
| | ASA $\alpha = 0.35$ | 13.68 | 2.95 | 14.91 | 16.38 | 3.85 | 12.52 | 17.39 | 4.23 | 11.67 |
| | ASA $\alpha = 0.40$ | 13.75 | 2.97 | 16.84 | 16.48 | 3.88 | 14.31 | 17.51 | 4.29 | 13.44 |
| | ASA $\alpha = 0.45$ | 13.85 | 2.99 | 18.58 | 16.61 | 3.92 | 16.15 | 17.65 | 4.32 | 15.39 |
| | ASA $\alpha = 0.50$ | 13.94 | 3.02 | 20.02 | 16.72 | 3.93 | 17.96 | 17.77 | 4.36 | 17.13 |
| | ASA $\alpha = 0.525$ | 14.01 | 3.06 | 21.15 | 16.83 | 3.95 | 19.55 | 17.82 | 4.37 | 18.69 |
| | ALAP | 14.27 | 3.11 | 26.25 | 17.02 | 4.00 | 25.88 | 17.99 | 4.47 | 25.91 |
| | Static | 16.56 | 3.83 | – | 19.04 | 4.87 | – | 19.61 | 5.16 | – |
| *ADP* ($\varepsilon = -5$) | ASAP | 32.30 | −0.75 | 6.59 | 36.15 | −0.44 | 5.89 | 37.41 | −0.11 | 5.48 |
| | ASA $\alpha = 0.1$ | 32.60 | −0.81 | 6.74 | 36.00 | −0.22 | 6.23 | 37.41 | −0.07 | 5.57 |
| | ASA $\alpha = 0.20$ | 32.92 | −0.63 | 7.79 | 36.10 | 0.05 | 6.81 | 37.54 | 0.28 | 6.23 |
| | ASA $\alpha = 0.25$ | 33.22 | −0.49 | 9.28 | 36.37 | 0.23 | 8.01 | 37.74 | 0.51 | 7.18 |
| | ASA $\alpha = 0.30$ | 33.85 | −0.38 | 12.34 | 37.00 | 0.32 | 10.49 | 38.20 | 0.64 | 9.39 |
| | ASA $\alpha = 0.35$ | 34.31 | −0.39 | 18.18 | 37.59 | 0.31 | 15.93 | 38.79 | 0.62 | 15.01 |
| | ASA $\alpha = 0.40$ | 34.59 | −0.37 | 20.78 | 37.98 | 0.30 | 18.71 | 39.28 | 0.60 | 17.94 |
| | ASA $\alpha = 0.45$ | 34.79 | −0.41 | 22.80 | 38.27 | 0.28 | 20.89 | 39.57 | 0.58 | 20.40 |
| | ASA $\alpha = 0.50$ | 34.92 | −0.39 | 23.82 | 38.40 | 0.29 | 22.42 | 39.73 | 0.62 | 22.02 |
| | ASA $\alpha = 0.525$ | 34.73 | −0.26 | 24.15 | 38.62 | 0.25 | 23.16 | 39.78 | 0.62 | 22.70 |
| | ALAP | 34.86 | −0.22 | 26.26 | 38.81 | 0.32 | 25.97 | 39.96 | 0.72 | 25.96 |
| | Static | 39.29 | 0.08 | – | 42.53 | 0.63 | – | 42.76 | 0.99 | – |

(*continued on next page*)

**Table 2** (*continued*)

| Objective function | Dynamic matching policy | Participation rate | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | 0.25% | | | 0.5% | | | 0.75% | | |
| | | MR (%) | AKS (%) | AFT | MR (%) | AKS (%) | AFT | MR (%) | AKS (%) | AFT |
| *DS* | ASAP | 13.28 | 2.68 | 7.86 | 15.72 | 3.46 | 6.85 | 17.01 | 3.84 | 6.33 |
| | ASA $\alpha$ = 1 km | 13.20 | 2.81 | 9.68 | 15.71 | 3.60 | 8.63 | 16.64 | 3.99 | 8.15 |
| | ASA $\alpha$ = 2 km | 13.14 | 2.93 | 12.44 | 15.65 | 3.72 | 11.24 | 16.57 | 4.12 | 10.67 |
| | ASA $\alpha$ = 3 km | 13.15 | 3.01 | 14.34 | 15.66 | 3.83 | 13.25 | 16.54 | 4.23 | 12.68 |
| | ASA $\alpha$ = 4 km | 13.15 | 3.08 | 15.88 | 15.68 | 3.90 | 14.73 | 16.58 | 4.32 | 14.27 |
| | ASA $\alpha$ = 5 km | 13.23 | 3.14 | 17.51 | 15.75 | 3.98 | 16.34 | 16.63 | 4.40 | 15.90 |
| | ASA $\alpha$ = 6 km | 13.28 | 3.18 | 18.76 | 15.78 | 4.04 | 17.56 | 16.68 | 4.48 | 17.19 |
| | ASA $\alpha$ = 7 km | 13.31 | 3.22 | 19.87 | 15.82 | 4.10 | 18.55 | 16.74 | 4.55 | 18.19 |
| | ASA $\alpha$ = 8 km | 13.35 | 3.26 | 20.80 | 15.87 | 4.15 | 19.46 | 16.79 | 4.61 | 19.07 |
| | ASA $\alpha$ = 9 km | 13.37 | 3.28 | 21.56 | 15.91 | 4.19 | 20.33 | 16.81 | 4.65 | 19.91 |
| | ALAP | 13.50 | 3.43 | 26.19 | 16.06 | 4.44 | 25.62 | 17.02 | 4.95 | 25.73 |
| | Static | 15.47 | 4.21 | – | 17.78 | 5.26 | – | 18.63 | 5.76 | – |

between the ASAP and the ALAP policies. Second, these results demonstrate that the ALAP-based approaches can generate competitive results to static situations where all information is available. For instance, DP-ALAP provides on average 90% of the MR value achieved by the DP-Static; however, this policy does not perform well in terms of the AFT measure. From the AFT measure point of view, ASAP policies outperform ALAP policies by a factor of four. Further, in terms of both ASAP and ALAP policies, the NM and DS objective functions dominate each other based on MR and AKS values, respectively.

DP and ADP objective functions yield non-dominated solutions. Specifically, DP dominates ADP with regard to MR, while it is dominated by ADP with regard to AKS. These differences can be explained in part due to lack of the length of the matched trip consideration in the DP. In fact, DP results in outstanding MR values.

When the ADP index is used in the objective function, the ADP-ASA $\alpha$ (for a wide range of $\alpha$ values) policy significantly outperforms DS-ASA $\alpha$ policies in terms of the MR and AKS (see Fig. 7). Further, using DP and ADP as objective functions, we observe that increasing the flexibility of the dynamic matching policy, i.e. from ASAP, to ASA $\alpha$ and then to ALAP, improves MR and AKS. In contrast, the DS-ASA $\alpha$ policy does not steadily improve the MR compared to DP-ASA $\alpha$ and ADP-ASA $\alpha$ policies (see Fig. 7). Another notable observation is that DP-ASA $\alpha$ and ADP-ASA $\alpha$ policies rapidly converge to the AKS value obtained from DP-ALAP and ADP-ALAP combinations, respectively. This means DP-ASA $\alpha$ and ADP-ASA $\alpha$ policies can significantly improve AKS while maintaining AFT at an acceptable level.

A closer inspection of the figure above shows that for both ASAP and ALAP dynamic matching policies, the DP and ADP objective functions outperform the NM objective function in terms of both MR and AKS measures. This suggests that even if maximizing MR is the main objective of the dynamic ride-sharing system, the DP and ADP indices would be desirable weighting strategies for the objective function of the matching problems in a dynamic, rolling-horizon based framework. The DP and ADP indices outperform the NM objective function since in comparison with a pure matching strategy, they prioritize announcements based on the announcements' original distances.

As mentioned by Stiglic et al. (2016), the spatial density of participants is important in ride-sharing systems. The same outcome is observed in Fig. 7; higher participation rate (and as a result, density) produces higher number of matches as well as higher distance savings.

Fig. 8 shows the impact of different objective functions across multiple participation rates. As can be perceived from the figure (and also Fig.7), with higher participation rates, ASA $\alpha$ policies result in better AFT. This means service providers can choose higher values of $\alpha$ to obtain higher improvement with no change in AFT of the system. Another remarkable issue is that DP-ASA $\alpha$ outperforms DP-ASAP and DP-ALAP in terms of AKS for many values for $\alpha$.

Comparing AFT values for ADP-ASA $\alpha$ for $\varepsilon = -5$ (Fig. 8) and $\varepsilon = 0$ (Fig. 7) reveals that the ADP objective function is sensitive to the value of $\varepsilon$. The former case, $\varepsilon = -5$, requires higher AFT for ADP-ASA $\alpha$ to converge to ADP-ALAP compared to DP-ASA $\alpha$.The computational performance of the ROLLING HORIZON algorithm is analysed in Fig. 9 for a range of time steps ($p$). We consider combinations of ASAP and ALAP matching policies, and different objective functions for the setting of participation rate = 0.50% and $\varepsilon = 0$. The computation time refers to the total time required for running the algorithms in all iterations. The results show that increasing the time step decreases the total computation time at the cost of decreasing the MR and the AKS criteria.

Next, we investigate the probability of finding a match based on participant's trip length. Fig. 10 breaks down trip announcements based on the participants' individual trip distance to investigate their success rate in finding a match. As shown in the figure, except for $\varepsilon = -5$ for DP, for all other cases, the probability of finding a match for drivers steadily increases with an increase in the trip length. Longer trips increase the probability of finding a compatible rider both en-route and detour which can then result in having more potential distance savings. For rider trips, we find that the probability of finding a match is highly dependent on the type of objective function and the value of $\varepsilon$. For example, using DP as the objective function, for rider trips shorter than 7 km a higher chance exists for $\varepsilon = -5$ while they have the lowest probability with $\varepsilon = 0$. Further, the likelihood of finding a match for a rider trip shorter than 7 km is low if DS is used in the objective function. Although short rider trips may easily be matched to compatible

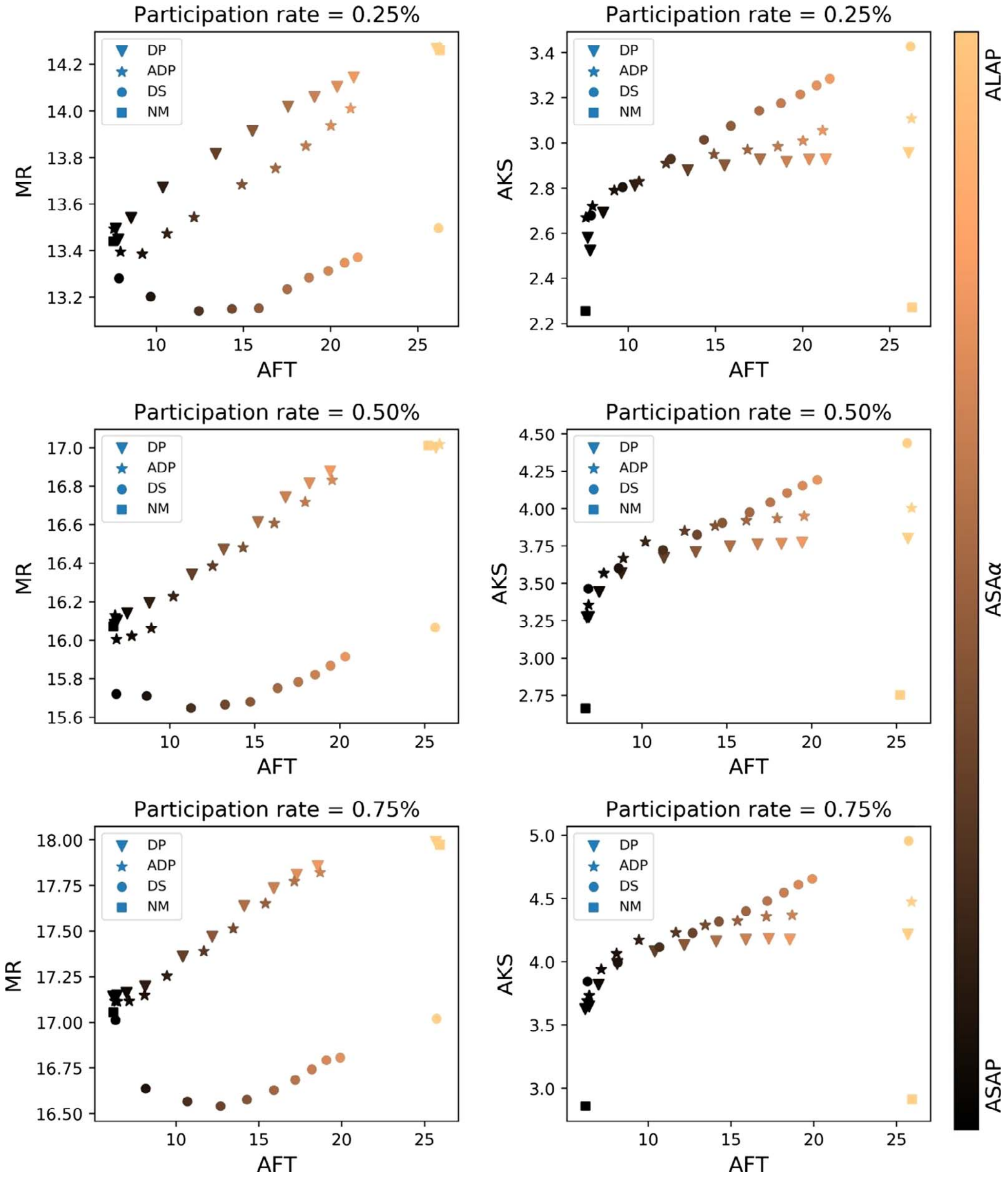



**Fig. 7.** Comparison of objective functions over different dynamic matching policies ($\varepsilon = 0$).

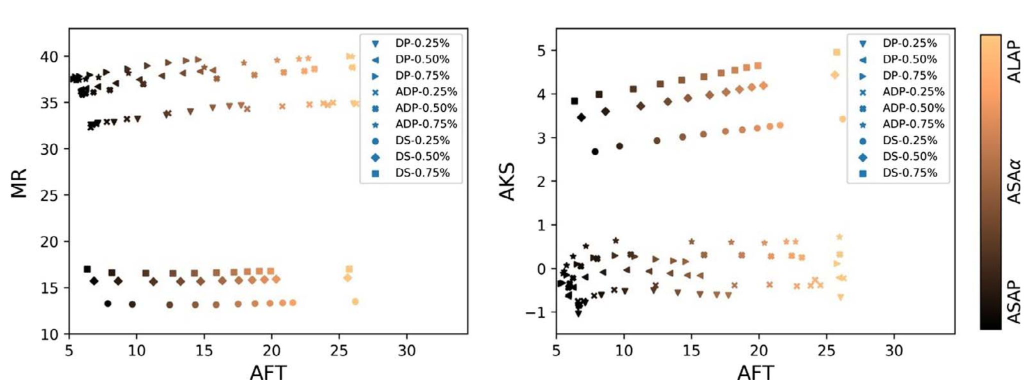
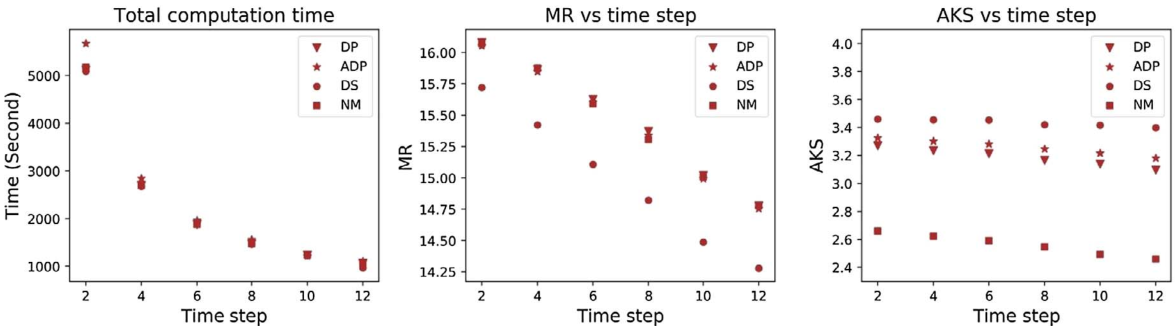
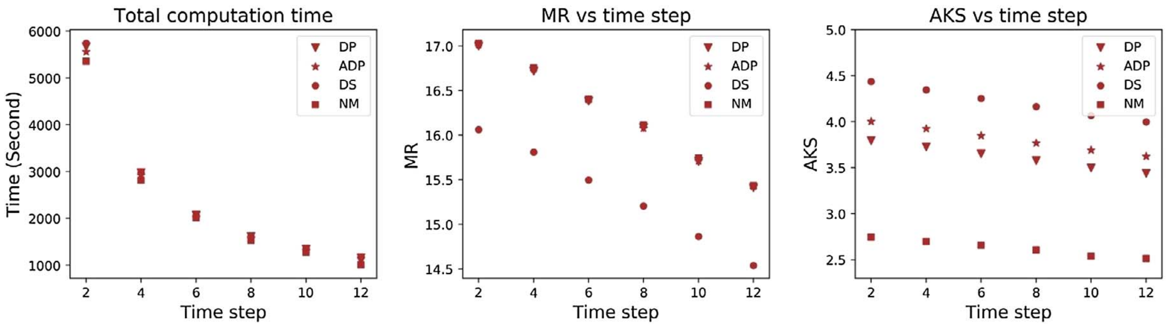



**Fig. 8.** Comparison of objective functions over different dynamic matching policies and participation rates ($\varepsilon = -5$).




a) Matching policy is ASAP




b) Matching policy is ALAP

**Fig. 9.** The effects of time step on the performance of the system ($\varepsilon = 0$ and participation rate = 0.50%).

drivers, they are unlikely to result in significant distance savings. Finally, we find that a higher trip density usually increases the probability of finding a match.

### *6.3. Performance of the clustering heuristic*

In this section, we analyse the performance of the proposed clustering heuristic for the presented ride-sharing system. To compare the results obtained from the CLUSTERING algorithm with our exact approach, we focus on participation rate of 0.5% and the ALAP policy, which is the setting that performed the worst in terms of computation time. In addition, we set $\varepsilon = -5$, $p = 2$; and the ROLLING HORIZON algorithm is executed over 450 time periods. To explore the sensitivity of the heuristic with regards to the number of clusters, we conduct experiments with $|N|= 2$, 3, 4 and 5 clusters.

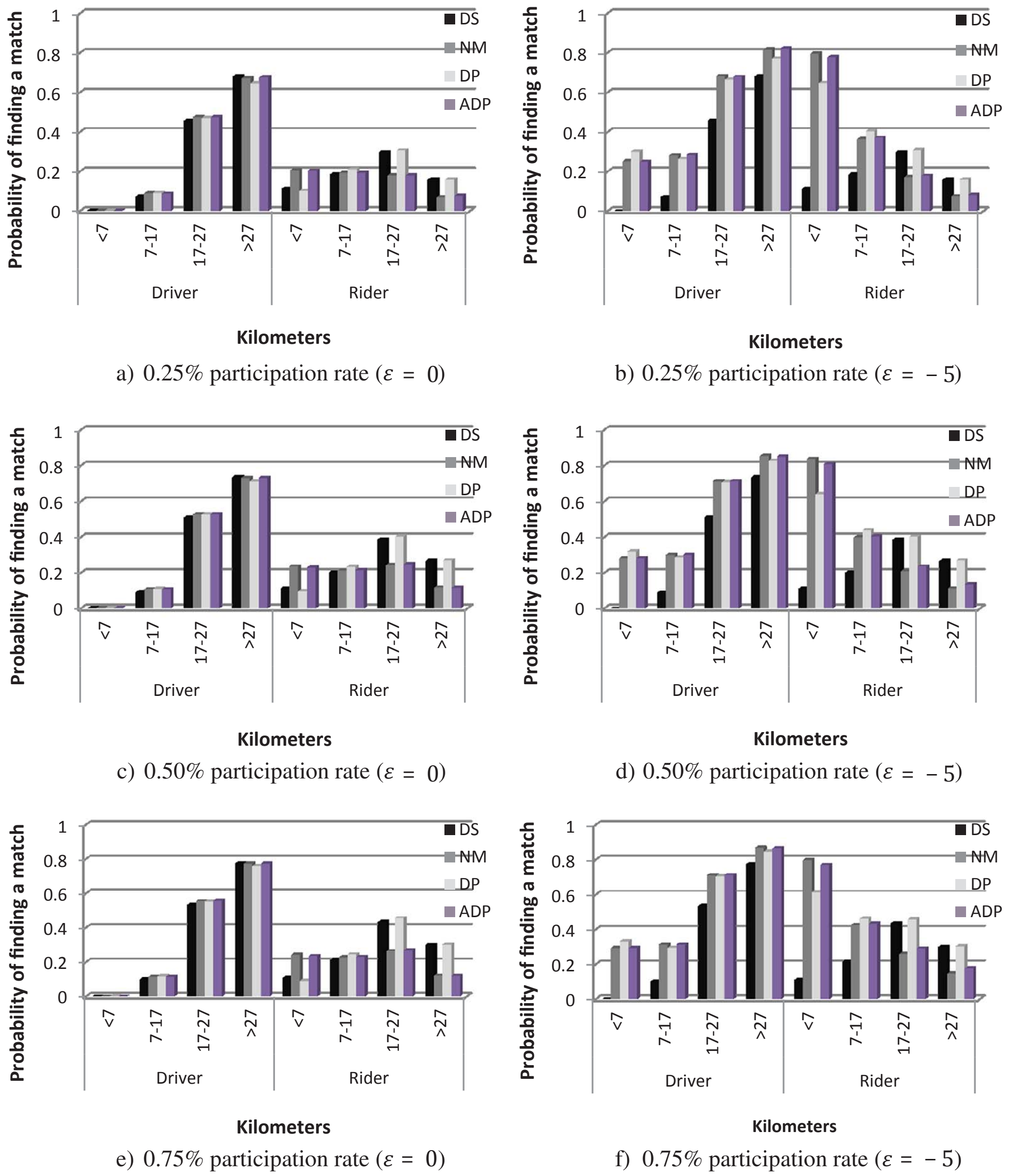
The results are shown in Table 3. The computation time values for both the STATIC and ROLLING HORIZON algorithms are provided in the 3rd an 4th columns for each number of clustered considered. It is notable that the matching algorithm is included under ROLLING HORIZON in the table. The average runtime per iteration is given in the 5th column. In addition, the performance of the algorithm with regards to MR, AKS and AFT is provided in the 6th, 8th and 9th columns, respectively. Further, the cumulative loss of MR for each number of clustered considered in the 7th column.

According to the results, the ROLLING HORIZON algorithm together with the NM objective function is significantly faster where the clustering algorithm does not affect the computation time of the algorithm. However, the clustering algorithm is considerably effective for reducing computation time for the STATIC algorithm. For DP, ADP and DS cases, using the clustering heuristic significantly reduces the total computational time by 44.3–68.8%, 50.8–75% and 45.3–70.7% respectively. The impact on the MR translates into reductions within the range of 1.4–2.9% in the case of two clusters, 2.1–3.9% in the case of three clusters, 2–4% in the case of four clusters, and 3–4.9% in the case of five clusters. With regard to the AKS, while the clustering algorithm does not notably affect the quality of the solution for DP and NM, this criterion is more influenced if ADP or DS are used. The AKS quality reduction for DS is within the range of 1.6–7%. This shows that the overall impact of the clustering heuristic is marginal compared to the quality of solutions.

## 7. Conclusion

### *7.1. Summary of findings*

In this paper, we proposed a new on-demand ride-sharing system, including novel objective functions for the matching problem therein, new dynamic matching policies and a clustering heuristic to tackle large-scale instances. We found that the choice of the objective function and the dynamic matching policy can substantially improve the performance of the ride-sharing system. The




a) 0.25% participation rate ($\varepsilon = 0$)

b) 0.25% participation rate ($\varepsilon = -5$)

c) 0.50% participation rate ($\varepsilon = 0$)

d) 0.50% participation rate ($\varepsilon = -5$)

e) 0.75% participation rate ($\varepsilon = 0$)

f) 0.75% participation rate ($\varepsilon = -5$)

**Fig. 10.** Probability of finding a match vs. participants' individual trip distance.

results highlighted that the proposed objective functions DP and ADP are able to outperform the traditional NM objective function. However, combination of DP, ADP and DS objective functions with ASAP and ALAP matching policies generate a set of non-dominated solutions so that none of the approaches is absolutely better than the other. We also reported that the ASA $\alpha$ policies are not compatible with the DS objective function (compared with DP and ADP objective functions). Further, we found that the results obtained using Policy DS-ASA $\alpha$ outperform both Policies DP-ASA $\alpha$ and ADP-ASA $\alpha$ with regards to MR. However, we observed that DP and ADP objective functions are well compatible with ASA $\alpha$ policies so that DP-ASA $\alpha$ and ADP-ASA $\alpha$ combinations outperform DS-ASA $\alpha$ in some AFT values. Moreover, the combinations can result in MR and AKS values at acceptable levels while their AFT measure has a significant difference to those of DP-ALAP, ADP-ALAP and DS-ALAP.

### *7.2. Policy implications*

Although ride-sharing is a promising and competitive approach to replace private car ownership, its success (quality of solutions) is closely tied to patronage. The quality of solutions assessed based on the performance measures evaluated is highly correlated with

**Table 3**
Performance of clustering algorithm (policy = ALAP, $\varepsilon = -5$, $p = 2$, and participation rate = 0.50%).

| Objective function | No. of clusters | STATIC computation time (seconds) | ROLLING HORIZON computation time (seconds) | Average runtime per iteration | MR | Cumulative loss of MR performance (%) | AKS | AFT |
|---|---|---|---|---|---|---|---|---|
| DP | 1 | 5440.27 | 1806.67 | 16.10 | 38.78 | 100 | −0.20 | 25.77 |
| | 2 | 2934.56 | 1101.08 | 8.97 | 38.09 | 98.22 | −0.16 | 26.02 |
| | 3 | 2550.55 | 1023.26 | 7.94 | 37.95 | 97.86 | −0.21 | 26.12 |
| | 4 | 1756.81 | 1100.28 | 6.35 | 37.98 | 97.94 | −0.31 | 26.14 |
| | 5 | 1398.75 | 865.81 | 5.03 | 37.62 | 97.01 | −0.30 | 26.26 |
| ADP | 1 | 5575.23 | 2534.95 | 18.02 | 38.84 | 100.00 | 0.37 | 25.88 |
| | 2 | 2763.29 | 1227.74 | 8.87 | 37.96 | 97.74 | 0.05 | 26.07 |
| | 3 | 2074.88 | 1021.47 | 6.88 | 37.78 | 97.28 | 0.03 | 26.17 |
| | 4 | 1798.89 | 1061.21 | 6.36 | 37.92 | 97.62 | −0.21 | 26.12 |
| | 5 | 1244.63 | 782.29 | 4.50 | 37.36 | 96.19 | −0.28 | 26.21 |
| DS | 1 | 5337.77 | 1314.25 | 14.78 | 16.06 | 100 | 4.44 | 25.73 |
| | 2 | 2881.73 | 754.04 | 8.08 | 15.84 | 98.63 | 4.37 | 25.98 |
| | 3 | 2486.79 | 709.29 | 7.10 | 15.68 | 97.63 | 4.28 | 26.08 |
| | 4 | 1749.79 | 733.96 | 5.52 | 15.42 | 96.01 | 4.20 | 26.29 |
| | 5 | 1376.37 | 575.57 | 4.34 | 15.28 | 95.14 | 4.13 | 26.34 |
| NM | 1 | 5321.20 | 18.99 | 11.87 | 38.73 | 100 | −1.44 | 25.83 |
| | 2 | 2850.37 | 18.62 | 6.38 | 37.59 | 97.06 | −1.40 | 25.84 |
| | 3 | 2302.29 | 20.52 | 5.16 | 37.22 | 96.10 | −1.43 | 25.92 |
| | 4 | 1772.98 | 21.43 | 3.99 | 37.46 | 96.72 | −1.45 | 25.96 |
| | 5 | 1458.28 | 18.97 | 3.28 | 36.93 | 95.35 | −1.39 | 26.05 |

the participation rate. Therefore, incentive mechanisms aiming at encouraging people to participate in ride-sharing system can facilitate the development of mobility options that are as flexible as owning a car while being more sustainable. Financial supports for institutionalizing the use of ride-sharing options can be an immediate means to improve the market uptake of such emerging mode of transport.

Other than financial and cultural support, the performance of a ride-sharing system is another important factor that motivates people to use the ride-sharing mode. A smart selection of dynamic matching policies and objective functions, which are defined to reflect the preferences of the parties involved, can substantially improve the performance of the ride-sharing system, as discussed in the results section. For example, if cost reduction is the main objective of a ride-sharing system, minimizing the total distance travelled by postponing the matching decision as late as possible results in the best design. This objective might be of interest to city planners who are interested in reducing vehicle miles travelled which is highly correlated with fuel consumption, air pollution and traffic congestion in metropolitan areas. Similarly, a private commercial company is most likely interested in maximizing the matching rate to increase participation and the overall fares collected. In this scenario, participants should be fully satisfied with the service they receive; otherwise, they may lose their interest in this travel options. If the target participants are full-time workers with hard constraints on arrival times the early notification of successful matches is a critical feature of the ride-sharing system. In this case, a dynamic matching mechanism that fulfils the objectives of the commercial company (maximizing the matching rate) as well as that of the drivers and riders is of utmost significance. In short, taking into account the competing nature of objective functions, users' preferences is crucial in developing a successful ride-sharing system.

Our analysis reveals that the proposed DP and ADP indices are appropriate objective functions in a ride-sharing system. The proposed indices tend maximize the matching rate while maintaining the total distance savings and early finalization time at a reasonable level. It also complies with finding matches as soon as possible. Therefore, if combined with this dynamic matching policy, this approach can offer a competitive alternative for commercial ride-sharing companies.

### *7.3. Future directions*

Other parties besides drivers and riders such as service providers, government, and environmental communities may have interests in or affecting the success of a ride-sharing system. For instance, the advantages of ride-sharing may encourage the government and environmental communities to introduce incentives to subsidise ride-sharing. Therefore, conceptual models are needed to define and evaluate the interrelationship and causal effects among the parties.

A promising extension of this study is to investigate methods to distribute travel costs of the trip between the driver and riders. A number of solutions are proposed in the literature including: a proportional allocation of the costs of the joint trip based on the distances of the separate trips; an even allocation of the costs of the joint trip; and a bidding system for the trip price of potential riders. Developing an equilibrium mechanism to model the interrelationship among drivers and riders in a competitive environment based on supply–demand relationships would be a new and attractive area of research. Such research can bridge the pricing or cost sharing gaps, especially for complications associated with cost sharing for multiple parties involved. The equilibrium price can be a base price for long term to reduce the underestimation of rider bids or overestimation of driver bids in the bidding based ride-sharing systems.

## Acknowledgement

The authors are very grateful to Mr. Amarin Siripanich for the fruitful exchange of opinions and constructive criticism.

## References

Agatz, N.A.H., Erera, A., Savelsbergh, M.W.P., Wang, X., 2011. Dynamic ride-sharing: a simulation study in metro Atlanta. Transp. Res. Part B, Methodol. 45 (9), 1450–1464.

Agatz, N.A.H., Erera, A.L., Savelsbergh, M.W.P., Wang, X., 2012. Optimization for dynamic ride-sharing: a review. Eur. J. Oper. Res. 223 (2), 295–303.

Amey, A., 2011. Proposed Methodology for Estimating Rideshare Viability Within an Organization: Application to the MIT Community. Transportation Research Board Annual Meeting 2011, Paper 11–2585.

Blattner, M., Zhang, Y.-C., Maslov, S., 2007. Exploring an opinion network for taste prediction: an empirical study. Physica A 373, 753758.

Code of Virginia, 1989. Title 43.2. Retrived from Motor vehicle Chapter 14: http://leg1.state.va.us/cgi-bin/legp504.exe?000 + cod + 46.2-1400.

Department of Transport, 2009. Victorian Integrated Survey of Travel and Activity 2007 (VISTA 07). http://www.transport.vic.gov.au/research/statistics/victorian-integrated-survey-of-travel-and-activity.

Edmonds, J., 1965a. Path, trees and flowers. Can. J. Math. 17, 449–467.

Edmonds, J., 1965b. Matching and a polyhedron with 0, l vertices. J. Res. N. B. S. B 69 (April–June), 125–130.

Edmonds, J., Karp, R.M., 1972. Theoretical improvements in algorithmic efficiency for network flow problems. J. ACM. Assoc. Comput. Mach. 19 (2), 248–264.

Ford, L.R., Fulkerson, D.R., 1956. Maximal flow through a network. Can. J. Math. 8, 399–404.

Furuhata, M., Dessouky, M., Ordóñez, F., Brunet, M.E., Wang, X., Koenig, S., 2013. Ridesharing: the state-of-the-art and future directions. Transp. Res. Part B: Methodol. 57, 28–46.

Gargiulo, E., Giannantonio, R., Guercio, E., Borean, C., Zenezini, G., 2015. Dynamic ride-sharing service: are users ready to adopt it? Procedia Manuf. 3, 777–784.

Ghoseiri, K., Haghani, A., Hamedi, M., 2011. Real-Time Rideshare Matching Problem. University of Maryland, Department of Civil and Environmental Engineering UMD-2009-05.

Guillaume, J.L., Latapy, M., 2006. Bipartite graphs as models of complex networks. Phys. A: Stat. Theor. Phys. 371, 795–813.

Hagberg, A.A., Schult, D.A., Pieter, J.S., 2008. Exploring network structure, dynamics, and function using NetworkX. In: Varoquaux, Gäel, Vaught, Travis, Millman, Jarrod (Eds.), Proceedings of the 7th Python in Science Conference (SciPy2008), Pasadena, CA USA, pp. 11–15.

Harold, W.K., 1955. The Hungarian method for the assignment problem. Naval Res. Logist. Quart. 2, 83–97.

Hopcroft, John E., Karp, Richard M., 1973. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. SIAM J. Comput. 2 (4), 225–231.

Kamar, E., Horvitz, E., 2009. Collaboration and Shared Plans in the Open World: Studies of Ride-sharing. In: Proceedings of the Twenty-First International Joint Conference on Artificial Intelligence (IJCAI), vol. 9, pp. 187–194.

Li, Z., Ierapetritou, M.G., 2010. Rolling horizon based planning and scheduling integration with production capacity consideration. Chem Eng Sci 65, 5887–5900.

Lloyd, Stuart P., 1982. Least squares quantization in PCM. IEEE Trans. Inf. Theory 28 (2), 129–137. http://dx.doi.org/10.1109/TIT.1982.1056489.

Mahajan, V., Muller, E., Bass, F., 1995. Diffusion of new products: empirical generalizations and managerial uses. Market. Sci. 14 (3), 79–88.

Masoud, N., Lloret-Batlle, R., Jayakrishnan, R., 2017. Using bilateral trading to increase ridership and user permanence in ridesharing systems. Transp. Res. Part E 102, 60–77.

Nourinejad, M., Roorda, M.J., 2016. Agent based model for dynamic ride-sharing. Transp. Res. Part C 64, 117–132.

Ou, Q., Jin, Y.-D., Zhou, T., Wang, B.-H., Yin, B.-Q., 2007. Power-law strength-degree correlation from resource-allocation dynamics on weighted networks. Phys. Rev. E 75 (2), 021102.

Parker, A., 2004. Unsustainable trends in the Australian census data for the journey to work in Melbourne and other cities in Victoria. In: 27th Australasian Transport Research Forum. Adelaide.

Pelzer, D., Xiao, J., Zehe, D., Lees, M.H., Knoll, A.C., Aydt, H., 2015. A partition-based match making algorithm for dynamic ridesharing. IEEE Trans. Intell. Transp. Syst. 16 (5), 2587–2598.

Pillac, V., Guret, C., Medaglia, A.L., 2012. An event-driven optimization framework for dynamic vehicle routing. Decis. Support Syst. 54 (1), 414–423.

Shen, B., Huang, Y., Zhao, Y., 2015. Dynamic Ridesharing. The SIGSPATIAL Special 7 (3), 3–10.

Stiglic, M., Agatz, N., Savelsbergh, M., Gradisar, M., 2016. Making dynamic ride-sharing work: The impact of driver and rider flexibility. Transp. Res. Part E 91, 190–207.

Stiglic, M., Agatz, N., Savelsbergh, M., Gradisar, M., 2015. The benefits of meeting points in ride-sharing systems. Transp. Res. Part B: Methodol. 82, 36–53.

Veitch Lister Consulting (VLC), Zenith User Manual, Travel Demand Forecasting and Transport Infrastructure Planning, Brisbane, Australia, 2013.

Winter, S., Nittel, S., 2006. Ad-hoc shared-ride trip planning by mobile geosensor networks. Int. J. Geogr. Inf. Sci. 20 (8), 899–916.

Xing, X., Warden, T., Nicolai, T., Herzog, O., 2009. Smize: a spontaneous ride-sharing system for individual urban transit. In: Proceedings of the 7th German Conference on Multiagent System Technologies, MATES'09. Springer-Verlag, Berlin Heidelberg, pp. 165–176.

Galil, Zvi., 1986. Efficient algorithms for finding maximum matching in graphs. ACM Comput. Surv.