

A new fuzzy logic approach to capacitated dynamic Dial-a-Ride problem

Maher Maalouf^{a,*}, Cameron A. MacKenzie^b, Sridhar Radakrishnan^c, Mary Court^d

^a Industrial and System Engineering, Khalifa University, P.O. Box 127788, Abu Dhabi, United Arab Emirates

^b Defense Resources Management Institute, Naval Postgraduate School, Monterey, CA 93943, USA

^c School of Computer Science, University of Oklahoma, Norman, OK 73019, USA

^d School of Industrial Engineering, University of Oklahoma, Norman, OK 73019, USA

Received 31 December 2012; received in revised form 6 March 2014; accepted 27 March 2014

Available online 3 April 2014

Abstract

Almost all Dial-a-Ride problems (DARP) described in the literature pertain to the design of optimal routes and schedules for n customers who specify pick-up and drop-off times. In this article we assume that the customer is mainly concerned with the drop-off time because it is the most important to the customer. Based on the drop-off time specified by the customer and the customer's location, a pick-up time is calculated and given to the customer by the dispatching office. We base our formulation on a dynamic fuzzy logic approach in which a new request is assigned to a vehicle. The fuzzy logic algorithm chooses the vehicle to transport the customer by seeking to satisfy two objectives. The first reflects the customer's preference and minimizes the time a customer spends in the vehicle, and the second reflects the company's preference and minimizes the distance a vehicle needs to travel to transport the customer. The proposed heuristic algorithm is relatively simple and computationally efficient in comparison with most deterministic algorithms for solving both small and large sized problems.

© 2014 Elsevier B.V. All rights reserved.

Keywords: Dial-a-Ride problem; Fuzzy logic; Transportation

1. Introduction

Dial-a-Ride transportation services are employed in various areas, either to serve elderly people or people with disabilities. People call Dial-a-Ride services to request transportation to and from a location, and the Dial-a-Ride company tells the customer when a van or car will come to pick up the customer. The request can be made a few days in advance, or it could be made just a few hours before the customer would like to be picked up. The objective then for the Dial-a-Ride company is to design a set of minimum cost vehicle routes capable of accommodating all requests

* Corresponding author.

E-mail addresses: maher.maalouf@kustar.ac.ae (M. Maalouf), camacken@nps.edu (C.A. MacKenzie), sridhar@ou.edu (S. Radakrishnan), mcourt@ou.edu (M. Court).

under constraints. In addition, the company must decide which vehicle should pick up a customer and the pick-up and drop-off time for each customer. Thus, both the transportation costs and customer convenience must be weighted when making the routing decisions.

Dial-a-Ride problems (DARP) are classified into static DARP and dynamic DARP. Static DARP are concerned with finding optimal routes for vehicles based on requests made in advance by customers, and static DARP make decisions about vehicles and pick-up and drop-off times with full knowledge about customers' requests for a particular day. Dynamic DARP are concerned with requests made in real time when vehicles already have scheduled pick-ups and drop-offs. Both the static and dynamic types are employed in either a single-vehicle or a multi-vehicle DARP systems [1]. Although many DARP solutions assume a deterministic problem, uncertainty or imprecision is an important consideration. There may be uncertainty (or imprecision) over when vehicles arrive at their stops, and customers may express flexibility about when they want to be picked up and dropped off.

For the static multi-vehicle case, Jaw et al. [2] propose one of the first insertion heuristic algorithms to deal with multi-vehicle DARP using convex optimization techniques. The algorithm considers two types of customers: desired drop off time (DDT) specified by customers, and desired pick-up time (DPT) specified by customers. Cordeau [3] proposes a Branch-and-Cut algorithm for small to medium size instances. Cordeau and Laporte [4] propose a Tabu search algorithm for the static DARP. Diana and Dessouky [5] present a parallel regret insertion heuristic to solve DARP for large-scale systems with time windows. The computational results show the effectiveness of this approach by trading off solution quality and computational times. Zidi et al. [6] propose a mathematical model for multi-objective static DARP.

For the dynamic multi-vehicle case, Madsen et al. [7] solve a real-life problem in which customers can specify either a pick-up or drop-off time window. The algorithm is based on that of Jaw et al. [2] and dynamically inserts new requests into vehicle routes. Hyttiä et al. [8] develop an efficient and modular simulation to study the performance of large scale systems under specified trip arrival processes with a number of policies. In addition to that, they demonstrate the viability of dynamic DARP and the trade-off between work conducted and quality of service. Muñoz-Carpintero et al. [9] solve a dynamic vehicle routing problem using a hybrid predictive control strategy. They use genetic algorithms to efficiently solve the problem and fuzzy logic for demand prediction. Núñez et al. [10] present a hierarchical multi-objective model based on predictive control approach to solve a dynamic pickup and delivery problem. The claim is that the hierarchical nature of the system breaks down the optimization problem into manageable sub-problems. Kergosien et al. [11] employ a Tabu search algorithm iteratively to solve the dynamic version of assigning ambulance vehicles to move patients between care centers in a hospital complex given certain constraints.

Colorni and Righni [12] consider solving an objective function consisting of maximizing the number of customers served, optimizing the perceived service level, and minimizing the total distance traveled by the vehicles. Attanasio et al. [13] use Tabu search algorithm for both static and dynamic multi-vehicle DARP. Melachrinoudis et al. [14] propose a multi-objective optimization technique to solve DARP involving a health center. Bergvinsdottir et al. [15], Jorgensen et al. [16], and Cubillos et al. [17] use genetic algorithm (GA) to solve DARP. Herbawi et al. [18] also propose a genetic and insertion heuristic algorithm to solve dynamic DARP. De Paepe et al. [19] design a computer program (DARCLASS) which takes lists of problems with known complexity as inputs and generates a classification of all problems into three classes as output: solvable in polynomial time, NP-hard, or open. Coja-Oghlan et al. [20] show that even though the DARP is NP-hard when the network forms a caterpillar (worst-case), it can be solved efficiently (on average). Jun [21] proposes an insertion heuristic-based hybrid genetic algorithm to solve the vehicle sharing problem as a fuzzy vehicle routing problem. Van Lon et al. [22] create a decentralized multi-agent system to solve the dynamic Dial-a-Ride problem.

Fuzzy numbers can express not only uncertainty and imprecision, but flexible constraints as well [23]. Fuzzy modeling has the advantage that it provides an easy way to determine which vehicle is optimal for picking up a customer that reflects the realities of how dispatchers naturally think about the problem. Kikuchi [24] implements a fuzzy modeling approach to DARP by considering the desired time of vehicle arrival and travel time between two nodes as fuzzy numbers. Teodorovic and Radivojevic [25] use fuzzy logic to minimize the total traveling distance, vehicle waiting times, and passenger ride times by developing two approximating algorithms, one for vehicle selection and the other for the optimal insertion of DDT.

In this paper, we extend the uncapacitated FDARP algorithm, proposed by Maalouf et al. [26] to include a capacitated dynamic fuzzy modeling algorithm for DARP, similar to the one proposed by Teodorovic and Radivojevic [25]. Unlike the one proposed by Teodorovic and Radivojevic [25], the FDARP in this paper assumes that customers are

mainly concerned with the drop-off time instead of the pick-up time, and hence the system specifies the pick-up time. In addition, instead of using two approximate reasoning algorithms in the fuzzy optimization step, only one approximate reasoning algorithm is needed in the proposed method. Another important modification is that we allow a vehicle to pick up a customer and make another stop to pick up or drop off a different customer before dropping off the first customer.

The organization of the paper is as follows. Section 2 states the problem and Section 3 describes the algorithm. Details of the method and results are presented in Section 4. The conclusion is stated in Section 5.

2. Problem statement

The dynamic Dial-a-Ride system for which FDARP is designed assumes that customers specify DDTs. These customers rely on the Dial-a-Ride company to provide them with the best possible pick-up time. Each customer calls and expresses a DDT, the time he or she would like to be dropped off at a given location. Although the customer-provided DDT may be exact, the FDARP algorithm assumes the customer is willing to be dropped off earlier than DDT but no earlier than the earliest drop-off time (EDT). The customer's desired drop-off time is the latest drop-off time at which the customer will be dropped off.

In a dynamic algorithm, vehicles have already been assigned routes for picking up and dropping off customers. The fundamental problem is to assign a vehicle for the new request and indicate the pick-up time for the customer. In addition to satisfying the customer's request by dropping the customer off within the time window of what the customer desires, the algorithm seeks to minimize the time the customer spends in the vehicle (dwell time) and the distance traveled by a vehicle. These two objectives may conflict with each other, and as detailed below, we use fuzzy logic to trade off between these two objectives. This bi-objective problem incorporates both the preferences of the customers, who want to ride in the vehicle as short a time as possible, and the preferences of the company, which desires efficient use of its vehicles. Many DARP solutions (see [4,25]) exclusively focus on vehicle distance, time, or cost, but companies also want to provide good customer service [7]. One way to reflect good customer service is to minimize the time customers spend in a vehicle. We assume that all vehicles originate from one depot and that all vehicles have the same capacity and are of the same type. Once a request is assigned, it is never modified.

The algorithm uses this objective to determine the actual pick-up and drop-off times for each customer. Given a list of N customers, each specifying EDT_i and DDT_i ($i = 1, 2, \dots, N$), the following relations must hold when determining the actual drop-off time, ADT_i , and the actual pick-up time, APT_i , where DRT_i is the direct ride-time between pick-up and drop-off and MDT_i is maximum dwell time that the customer is willing to spend in the vehicle:

$$DDT_i \geq ADT_i \geq EDT_i, \quad (1)$$

$$MDT_i \geq ADT_i - APT_i \geq DRT_i. \quad (2)$$

Relation (1) states that the actual drop-off time must be between the earliest drop-off time and the desired drop-off time. Relation (2) indicates that the actual pick-up time subtracted from actual drop-off time must be at least as large as the direct time it takes to go from pick-up to drop-off, and that difference must not be greater than the maximum time the customer is willing to spend in the vehicle. The maximum dwell-time MDT_i could either be stated by the customer, but more likely it will be predefined by the Dial-a-Ride company that a customer will not spend more than a certain amount of time in the vehicle. This allows the company to pick up a passenger and make another stop before dropping off that the same customer and differs from the model proposed by Teodorovic and Radivojevic [25].

3. Proposed solution

The proposed solution is composed of two major parts. The first part utilizes fuzzy numbers to determine when a vehicle could pick up and drop off a potential customer. The second part deploys a fuzzy logic system or an approximate reasoning algorithm to determine the optimal vehicle to pick up the customer. As mentioned earlier, fuzzy numbers express imprecision. While most engineering design problems depend on *objective knowledge*, representing mathematical models and equations, many real-life problems involve *subjective knowledge*, representing linguistic information that is difficult to quantify precisely [27]. In traffic and transportation engineering, subjective knowledge

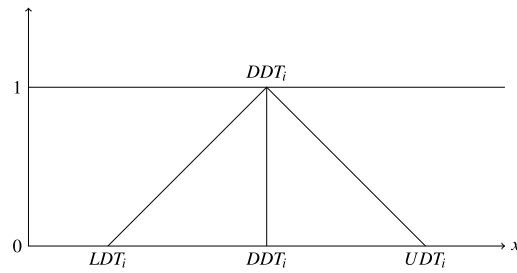


Fig. 1. Desired drop-off time represented as a triangular fuzzy number.

and linguistic information are used on a daily basis by drivers, dispatchers, and passengers [28,29]. Subjective statements such as “reservations are made with a forty-minute pickup window” or “travel time is approximately twenty five minutes” are often used. Thus, the use of fuzzy models to solve such problems is very suitable.

Fuzzy logic systems are composed of the following three steps: fuzzification, rule evaluation, and defuzzification. The fuzzification step is based on converting inputs into membership degree (between 0 and 1) based on the membership functions. Usually membership functions are triangular (as the one used in this paper), trapezoidal, or Gaussian. Rule evaluation is based on ‘If-Then’ statements, combining membership functions to derive another fuzzy output. Defuzzification then calculates a single numerical value based on the inferred results from the rules, and the center-of-gravity method (centroid) is usually used. Interested readers are referred to Kaufmann and Gupta [30], Mendel [27], and Zimmerman [32] for rigorous mathematical details.

Teodorovic and Radivojevic [25] solve a dynamic DARP using fuzzy logic with two approximate reasoning algorithms, one for vehicle selection, and the other for the optimal insertion of DDT . Their approximating algorithms are aimed at customers with a specified pick-up time, DPT . Unlike their algorithm, the proposed algorithm for this paper is designed for customers with a specified drop-off time, DDT , because we believe more emphasis should be given to DDT .

Customers are usually more concerned with the drop-off time than with the pick-up time. In addition, rather than allowing the dispatcher’s subjective perception as to which vehicle to dispatch for a certain customer, FDARP can automatically provide such information. Furthermore, instead of using two approximate reasoning algorithms in the fuzzy optimization step, only one fuzzy approximate reasoning algorithm is needed in the proposed method.

The methodology in this paper consists of two parts. A request for a Dial-a-Ride is made. In the first part, an algorithm determines the best pick-up and drop-off time for each vehicle in the fleet. Certain operational constraints must be satisfied, including vehicle capacity, the maximum ride time, and the customer’s drop-off time. After a pick-up and drop-off time for the request is determined for each vehicle, an optimization algorithm in the second part based on fuzzy logic determines the best vehicle to satisfy the customer’s request. The fuzzy optimization algorithm seeks to minimize both the amount of time a customer stays in the vehicle and the distance traveled by the vehicle. The proposed solution is heuristic and generally does not generate an optimal solution in the deterministic operations research sense. In addition, it is assumed here that once a customer node is inserted into the schedule, it remains unchanged.

3.1. Part 1: pick-up and drop-off times for each vehicle

The first step determines the best time that each of the M vehicles in the fleet can pick up and drop off customer i , who has specified DDT_i and EDT_i . Following Teodorovic and Radivojevic [25], we model the desired drop-off time as a triangular fuzzy number to represent the uncertainty in the customer’s request. For example, we believe a customer will say that he or she would like to be dropped off around 8:00, as opposed to exactly at 8:00. A fuzzy number $DDT_i = (LDT_i, DDT_i, UDT_i)$, where LDT_i and UDT_i are the lower and upper bounds on the fuzzy number, is shown in Fig. 1.

FDARP sorts all pick-up or drop-off times for a given vehicle in ascending order. Since DDT_i is the input to FDARP, the insertion of the drop-off node i^- occurs before the insertion of pick-up node i^+ . A non-trivial problem emerges when these two nodes must be separated by another node, where each node represents a stop for a vehicle.

Algorithm 1: Algorithm to determine pick-up and drop-off time for customer i .

Data: M, DDT_i
Result: dw, dis

```

1 begin
2   for  $j \leftarrow 1$  to  $M$  do
3     node  $e$  = stop immediately preceding  $i^-$ 
4     node  $f$  = stop immediately succeeding  $i^-$ 
5     while drop-off time  $j$  unassigned do
6       if  $AA_f - AA_e > TT_{ei^-} + TT_{i^-f}$  then
7         drop-off time  $j$  =  $\min(DDT_i, AA_f - TT_{i^-f})$ 
8         if  $AA_f - AA_e > TT_{ei^+} + TT_{i^+i^-} + TT_{i^-f}$  then
9           pick-up time  $j$  = drop-off time  $j$  -  $TT_{i^+i^-}$ 
10        else
11          node  $\ell$  = node  $e$ 
12          node  $k$  = node preceding  $\ell$ 
13          while pick-up time  $j$  unassigned do
14            if  $AA_\ell - AA_k > TT_{ki^+} + TT_{i^+\ell}$  then
15              pick-up time  $j$  =  $AA_\ell - TT_{i^+\ell}$ 
16              if capacity with customer  $i$  exceeds maximum capacity of vehicle  $j$  then
17                unassign drop-off time  $j$  and pick-up time  $j$ 
18                node  $f$  = node  $e$ 
19                node  $e$  = node preceding  $f$ 
20            else
21              node  $\ell$  = node  $k$ 
22              node  $k$  = node preceding  $\ell$ 
23        else
24          node  $f$  = node  $e$ 
25          node  $e$  = node preceding  $f$ 
26  max dwell time =  $\min(\text{pick-up time}_j \text{ for all } M \text{ vehicles})$ 
27  min dwell time =  $\max(\text{pick-up time}_j \text{ for all } M \text{ vehicles})$ 
28  max distance = maximum additional distance a vehicle travels to pick up and drop off customer  $i$ 
29  min distance = minimum additional distance a vehicle travels to pick up and drop off customer  $i$ 
30  dwell score  $dw_j$  = function of pick-up time  $j$ , min dwell time, and max dwell time; dwell score closer to 0 indicates that
    passenger's dwell time is small
31  distance score  $dis_j$  =  $(\text{additional distance vehicle } j \text{ travels to pick-up and drop-off customer } i - \text{min distance}) / (\text{max}$ 
    distance - min distance)

```

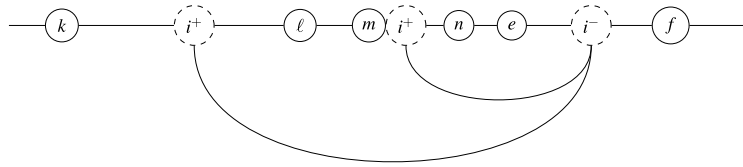
The following constraints must be satisfied for the actual drop-off time ADT_i :

$$ADT_i \geq AA_e + TT_{ei^-}, \quad (3)$$

$$AA_f \geq ADT_i + TT_{i^-f}, \quad (4)$$

where AA_e is the actual arrival time at node e (the node that precedes the drop-off node of customer i), TT_{ei^-} is the travel time from node e to node i^- , TT_{i^-f} is the travel time from node i^- to node f (the node that succeeds the drop-off node of customer i), and AA_f is the actual arrival time at node f .

The FDARP algorithm for this first part is outlined in Algorithm 1. The algorithm assumes that all times are expressed as fuzzy numbers, to include drop-off and pick-up times, travel times, and arrival times, and we use the fuzzy arithmetic methods described by Teodorovic and Vukadinovic [29] and Kaufmann and Gupta [31] to ensure

Fig. 2. Assignment of a new request i to a route of a vehicle j .

that these operational constraints are satisfied.¹ The algorithm can also work assuming that all numbers are precise. When one fuzzy number is subtracted from another fuzzy number in the algorithm, the fuzzy numbers are independent of each other [23]. The algorithm repeats for each of the M vehicles, with a computational complexity of $O(M)$.

Ideally, a vehicle should drop the customer off at the customer's desired drop-off time (DDT_i). If assigning the actual drop-off time (ADT_i) to be equivalent to DDT_i violates Eq. (3) or (4), the operator can change ADT_i to an earlier time. The actual drop-off time may need to be readjusted so that the customer is picked up before the current node e . If this occurs, the current node e becomes node f and the new node e immediately precedes the current node e .

After ADT_i is determined, the next step is the insertion of node i^+ , which requires the calculation of the actual pick-up time APT_i :

$$AA_k + TT_{ki^+} \leq APT_i, \quad (5)$$

$$APT_i + TT_{i^+\ell} \leq AA_\ell, \quad (6)$$

where AA_k is the actual arrival time at node k (the node preceding the pick-up node of customer i), TT_{ki^+} is the travel time from node k to node i^+ , $TT_{i^+\ell}$ is the travel time from node i^+ to node ℓ (the node that succeeds the pick-up node of customer i), and AA_ℓ is the actual arrival time at node ℓ .

The FDARP algorithm first assigns node i^+ such that $APT_i = ADT_i - DRT_i$. If this assignment violates Eq. (5) or (6), FDARP moves node i^+ to an earlier time, up to the MDT_i (Fig. 2). Regardless of whether a feasible solution is found for vehicle j , FDARP tries to find a drop-off and pick-up time for customer i for the next vehicle.

FDARP calculates a dwell score (dw_j) and a distance score (dis_j) for vehicle j . Both scores are functions of the assigned ADT_i and APT_i for each vehicle, and the scores are scaled so that each score ranges between 0 and 1. The dwell score is based on the length of time the customer spends in vehicle j . If the time spent in vehicle j is close to DRT_i , the dwell score is near 0. As the pick-up time becomes earlier for vehicle j , the dwell time increases, which means that dw_j moves towards 1. The distance score also ranges between 0 and 1 and is based on the distance vehicle j travels. The shorter the distance vehicle j travels, the closer the distance score is to 0. The FDARP algorithm ends by returning two vectors of length M : a vector of dwell scores and a vector of distance scores.

3.2. Part 2: selecting the optimal vehicle

Once feasible insertions for nodes i^- and i^+ are found, we use an approximate reasoning algorithm to select the optimal feasible solution. As mentioned earlier, optimality is defined as the strongest preference (very weak, weak, medium, strong, and very strong) based on the customer's dwell time and distance score for each vehicle. After the FDARP algorithm generates a dwell time dw_j and a distance score dis_j for each feasible vehicle j for one individual customer, determining which vehicle will carry the passenger is the next step.

A multiobjective value function, where the decision maker explicitly determines how much to weigh the distance a vehicle travels vis-a-vis the time a customer sits in the vehicle, could be used to determine the optimal vehicle. However, it may be unrealistic to expect that the decision maker will assess the weights and follow the normative rules of multicriteria decision making [33,34]. As an alternative, we deploy an approximate reasoning algorithm to trade off between the dwell score and the distance score. The approximate reasoning algorithm is based on Mamdani [35,36] fuzzy inference system as implemented by the Matlab [37] fuzzy logic toolbox. The Mamdani fuzzy inference

¹ The greater-than inequality $>$ in the algorithm is defined as follows. Let $A = (a_1, a_2, a_3)$ and $B = (b_1, b_2, b_3)$ be fuzzy numbers. $A > B$ if $(a_1 + 2a_2 + a_3)/4 > (b_1 + 2b_2 + b_3)/4$; or $(a_1 + 2a_2 + a_3)/4 = (b_1 + 2b_2 + b_3)/4$ and $a_2 > b_2$; or $(a_1 + 2a_2 + a_3)/4 = (b_1 + 2b_2 + b_3)/4$ and $a_2 = b_2$ and $a_3 - a_1 > b_3 - b_1$ [29,31].

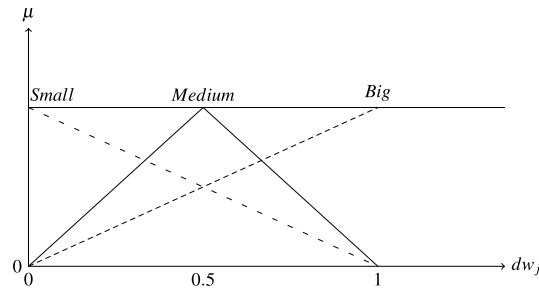


Fig. 3. Membership function that fuzzifies the dwell time of a new passenger.

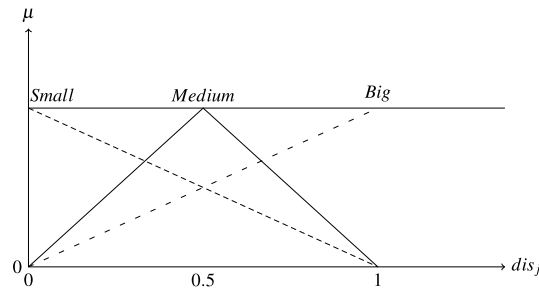


Fig. 4. Membership function that fuzzifies the distance traveled to pick up a new passenger.

Table 1

Algorithm 2: Fuzzy rules for choosing optimal vehicle.

		Dwell Time		
		<i>Small</i>	<i>Medium</i>	<i>Big</i>
Distance to Pick up	<i>Small</i>	Very Strong (VS)	Strong (S)	Medium (M)
	<i>Medium</i>	Medium (M)	Medium (M)	Weak (W)
	<i>Big</i>	Weak (W)	Weak (W)	Very Weak (VW)

system uses membership functions to fuzzify inputs; calculates an output based on rules that combine the fuzzified inputs; and classifies that output in a discrete category.

The membership functions in this paper are identical to those in Teodorovic and Radivojevic [25], and they transform the crisp scores dw_j and dis_j into a combination of “small”, “medium”, or “big” as depicted in Figs. 3 and 4. The triangular membership functions are designed to reflect a dispatcher’s subjective preference for a smaller dwell time and a shorter distance traveled. Each membership function assumes that a score of 0 is completely small (e.g., small dwell time or short distance traveled) and a score of 1 is completely big (e.g., large dwell time or long distance traveled). If the score equals 0.5, the resulting fuzzification is mostly medium with small and big at equal but lower levels. Although the numerical results use the membership functions depicted in Figs. 3 and 4, a user can modify these membership functions to better reflect the preferences of the dispatcher.

Because each score is partly small, partly medium, and partly big due to the membership function, the output is a distribution based on $3 \times 3 = 9$ relationships. Each of these relationships is depicted in Table 1. For example, **IF** the distance to pick up a new customer is *small* **AND** the dwell time is *small*, **THEN** the preference is **VERY STRONG**. Given these preference relationships and the previously described membership functions, the Matlab fuzzy toolbox calculates a numerical vehicle preference score ranging between 0 and 100. The vehicle preference score is categorized into one of five categories according to the structure in Fig. 5: very weak, weak, medium, strong, and very strong.

For example, if $dw_j = 0.2$ and $dis_j = 0.5$, the distance traveled by vehicle j to pick up a customer is mostly medium, and the customer is dropped off pretty close to his or her DDT, so the fuzzification of dw_j is mostly small. The Matlab fuzzy toolbox returns a numerical output of 50.9, which corresponds to a medium preference for that

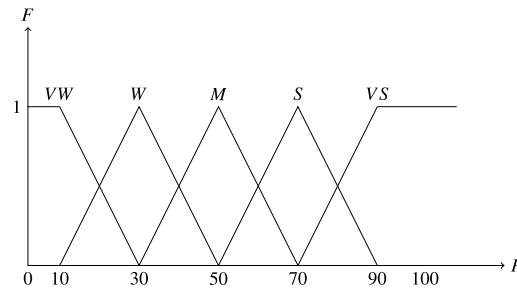


Fig. 5. Preference strength scale to assign a new request to a certain vehicle.

Table 2

Results for simulation with 6 vehicles and 100 customers.

Customers rejected because constraints not met any vehicle	7
Average time customer spends in vehicle	26.81 minutes
Average amount of time customer dropped off before DDT	2.32 minutes
Average number of miles traveled by a vehicle	305.05 miles
Total number of vehicles used	6
Average ratio of actual travel time to direct travel time	1.08–1.78
Average ratio of actual traveling distance to direct traveling distance	1.09–1.89
CPU elapsed time	4.37 seconds

vehicle (Fig. 5). Although a small dwell time and a medium distance correspond to a strong vehicle preference, the membership function for dw_j classifies a significant proportion of $dw_j = 0.2$ in medium and even big (Fig. 3). If the distance score is also 0.2, the numerical output is 61.5, and the vehicle preference is strong.

4. Numerical example

To the authors' knowledge, no benchmark data sets are available in the literature for the version of DARP studied in this paper. Therefore, the behavior of the proposed method cannot be evaluated using data instances that have been widely tested and the results cannot be compared to other optimum ones obtained in previous publications. Thus, in order to analyze the efficacy of the FDARP algorithm, we have simulated instances according to realistic assumptions. The performance measures used in this study, however, are very close to the ones used by Teodorovic and Radivojevic [25]. The developed algorithms above were tested using two different simulations for a single day. Both simulations start with a completely empty system. Customers call one at a time with a DDT_i , EDT_i , and LDT_i , and the algorithms determine the optimal vehicle to pick up and drop off customer i . Each DDT_i is randomly generated from a uniform distribution between 8 AM and 10 PM and $EDT_i = DDT_i - 10$ min and $LDT_i = DDT_i + 10$ min. The time it takes to go directly from picking the customer up to dropping him or her off, DRT_i , is uniformly distributed between 5 and 45 minutes. Each customer is willing to spend at most 1 hour in the vehicle, and each will also accept being dropped off at most 20 minutes earlier than his or her DDT_i . Consequently, a customer will be picked up at most 1 hour 20 minutes before DDT_i .

Although the dwell score is a fuzzy number, "small" dwell time corresponds approximately to less than 25 minutes, "medium" dwell time is more than 25 minutes but less than 50 minutes, and "big" dwell time is more than 50 minutes. The maximum distance a vehicle travels to pick up a customer is 25 miles, and small distance corresponds approximately to 0–8 miles, medium distance is 8–16 miles, and big distance is 16–25 miles. Each vehicle can hold a maximum of 10 people. All of the computations are carried out using MATLAB version 2012b on a 4 GB RAM computer.

In the first simulation, the Dial-a-Ride company has a fleet of 6 vehicles, and 100 customers request a ride. Table 2 displays the results of this simulation. Most customers arrive at their destination close to their DDT_i as they wait on average about 2 minutes although 7 customers are not given a ride because the operational constraints are not met. Customers being dropped off an average of only 2 minutes before their desired drop-off time is excellent customer service, and many customers in this simulation go directly from their pick-up to drop-off locations. However,

Table 3
Results for simulation with 30 vehicles and 900 customers.

Customers rejected because constraints not met any vehicle	26
Average time customer spends in vehicle	33.7 minutes
Average amount of time customer dropped off before DDT	4.47 minutes
Average number of miles traveled by a vehicle	414.63 miles
Total number of vehicles used	30
Average ratio of actual travel time to direct travel time	1.12–1.99
Average ratio of actual traveling distance to direct traveling distance	1.20–2.17
CPU elapsed time	213.3 seconds

rejecting 7 out of 100 customers because a vehicle cannot get them to their destination within the time window is a deficiency. Relaxing some constraints such as dropping customers off more than 20 minutes before their DDT_i or having customers spend more than 1 hour in the vehicle would allow the company to service more requests.

In the second simulation, the company has a fleet of 30 vehicles, and 900 customers request a ride. Table 3 displays the results of this simulation with a larger fleet of vehicles. Twenty-six customers are rejected, giving the company an ability to satisfy almost 98% of all requests. The service is a little less efficient than the smaller fleet with fewer customers, and customers arrive on average 4.5 minutes before their DDT_i , compared with 2 minutes in the first simulation, and the average customer spends 34 minutes in the vehicle, compared with 27 minutes in the first simulation. The average ratio of actual travel time to direct travel time is 1.12–1.99 and the average ratio of actual traveling distance to direct traveling distance is 1.20–2.17. These ratios are close to those obtained by Teodorovic and Radivojevic [25] who depicted values of 1.25–1.36 for actual travel time to direct travel time and 1.16–1.23 for actual traveling distance to direct traveling distance.

As can be observed from Tables 2 and 3, the CPU elapsed time for the algorithm to assign customers to vehicles is about 4 seconds for 6 vehicles and 100 customers and about 3.5 minutes for 30 vehicles and 900 customers. The time to process and find a vehicle for the next customer is also very short. When 100 customers are previously assigned to 6 vehicles, the algorithm assigns the 101st customer to a vehicle in 0.05 seconds on average. When 900 customers are previously assigned to 30 vehicles, the algorithm assigns the 901st customer to a vehicle in 0.25 seconds on average. This numerical study suggests that FDARP can find a solution in a short enough time period that a Dial-a-Ride company could rely on it in a real-time setting.

Varying the number of vehicles available for transporting customers can provide insight into how the size of the vehicle fleet impacts the performance characteristics. Fig. 6 depicts how the number of customers that can be served (out of a total of 900 customers) and the average miles traveled by each vehicle change as the number of vehicles in the fleet changes. The results are based on 5 simulation runs. As the vehicle fleet increases in size, the Dial-a-Ride company can serve more customers, and average distance traveled by each vehicle decreases. This type of comparison can be useful for a decision maker determining the size of the company's fleet. For example, if the company currently has 30 vehicles and is considering buying a 31st vehicle, a manager can use this comparison to understand that having a 31st vehicle will allow the company to transport about 9 more passengers each day and that each vehicle will travel on average 8 fewer miles per day. The manager can weigh these additional benefits with the extra cost of having an additional vehicle and driver.

If the company has 32 vehicles or more, the company is able to meet almost all of the customer requests. Depending on the cost of buying additional vehicles and hiring additional drivers, a company with 30 vehicles who fields 900 requests per day may want to consider investing in additional vehicles.

5. Conclusion

We have presented the FDARP algorithm and demonstrated its effectiveness for DARP. Basing the pick-up and drop-off times on the time a customer wishes to arrive at a location provides a realistic way of handling each customer's request. Allowing the vehicle to make multiple stops in between picking up and dropping off a customer gives a Dial-a-Ride greater flexibility. The fuzzy logic rules enables the Dial-a-Ride company to trade off between two objectives—one of which focuses on making the company more efficient and the other which focuses on customer service. By minimizing the distance traveled by each vehicle and attempting to drop off the customer at a time closest

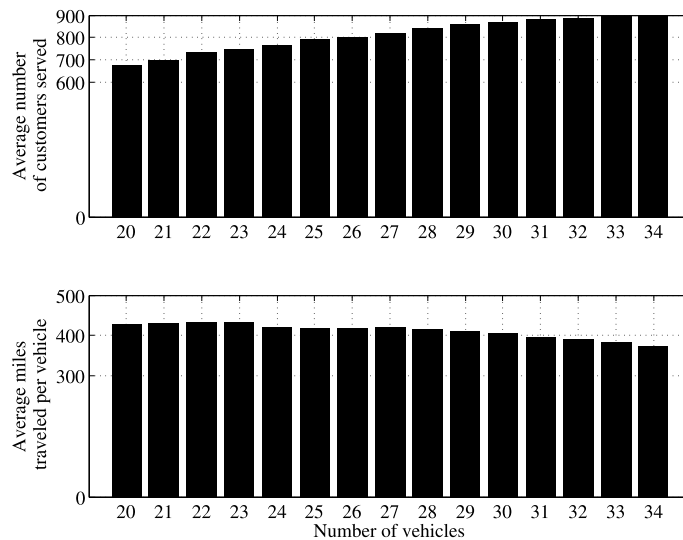


Fig. 6. Performance metrics for different vehicle fleet sizes with 900 customers.

to his or her desired drop-off time, the algorithm balances between a customer's preferences and a company's desire for efficiency and low cost.

FDARP takes advantage of the speed and flexibility of fuzzy rules to insert requests into a system, and it is less complex than algorithms that rely on constrained optimization methods. The numerical study provides insight into how simulation can help a company make decisions about the size of its vehicle fleet and which vehicle should pick up each customer. Using real-life data in future studies can demonstrate FDARP's usefulness especially if it is linked to a real-time GPS systems to maximize the algorithm's performance and accuracy. Although the algorithm is mainly concerned with dynamic DARP, it could be extended to solve static DARP and potentially other assignment problems.

Acknowledgements

The authors would like to thank Ms. Sarah Bawazir and Dr. Kamal Taha of Khalifa University for their valuable input, comments and suggestions.

References

- [1] J.-F. Cordeau, G. Laporte, The dial-a-ride problem: models and algorithms, *Ann. Oper. Res.* 153 (2007) 29–46.
- [2] J.-J. Jaw, A.R. Odoni, H.N. Psaraftis, A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows, *Transp. Res.* 20B (1986) 243–257.
- [3] J.-F. Cordeau, A branch-and-cut algorithm for the dial-a-ride problem, *Oper. Res.* 54 (3) (2006) 573–586.
- [4] J.-F. Cordeau, G. Laporte, A tabu search heuristic for the static multi-vehicle dial-a-ride problem, *Transp. Res., Part B* 37 (2003) 579–594.
- [5] M. Diana, M.M. Dessouky, A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows, *Transp. Res., Part B* 38 (2004) 539–557.
- [6] I. Zidi, K. Mesghouni, K. Zidi, K. Ghedira, A multi-objective simulated annealing for the multi-criteria dial a ride problem, *Eng. Appl. Artif. Intell.* 25 (6) (2012) 1121–1131.
- [7] O.B.C. Madsen, H.F. Ravn, J.M. Rygaard, A heuristic algorithm for a dial-a-ride problem with time windows, multiple capacities, and multiple objectives, *Ann. Oper. Res.* 60 (1995) 193–208.
- [8] E. Hyttiä, L. Häme, A. Penttinen, R. Sulonen, Simulation of a large scale dynamic pickup and delivery problem, in: *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques*, 2010, pp. 77:1–77:10.
- [9] D. Muñoz-Carpintero, A. Nuñez, D. Saez, C. Cortes, Evolutionary algorithms and fuzzy clustering for control of a dynamic vehicle routing problem oriented to user policy, in: *2010 IEEE Congress on Evolutionary Computation (CEC)*, 2010, pp. 1–8.
- [10] A. Nuñez, B. De Schutter, D. Sáez, C. Cortés, Hierarchical multiobjective model predictive control applied to a dynamic pickup and delivery problem, in: *Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems, ITSC 2010, Madeira Island, Portugal*, 2010, pp. 1553–1558.
- [11] Y. Kergosien, C. Lente', D. Piton, J.-C. Billaut, A tabu search heuristic for the dynamic transportation of patients between care units, *Eur. J. Oper. Res.* 214 (2011) 442–452.

- [12] A. Colomi, G. Righni, Modeling and optimizing dynamic dial-a-ride problems, *Int. Trans. Oper. Res.* 8 (2001) 155–166.
- [13] A. Attanasio, J.-F. Cordeau, G.G. Gilbert, Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem, *Parallel Comput.* 30 (2004) 377–387.
- [14] E. Melachrinoudis, A.B. Ilhan, H. Min, A dial-a-ride problem for client transportation in a health-care organization, *Comput. Oper. Res.* 34 (2007) 742–759.
- [15] K.B. Bergvinsdottir, J. Larsen, R.M. Jørgensen, Solving the dial-a-ride problem using genetic algorithms, Tech. rep., Informatics and Mathematical Modelling, Technical University of Denmark, DTU, 2004.
- [16] R.M. Jørgensen, J. Larsen, K.B. Bergvinsdottir, Solving the dial-a-ride problem using genetic algorithms, *J. Oper. Res. Soc.* 58 (2007) 1321–1331.
- [17] C. Cubillos, N. Rodríguez, B. Crawford, A study on genetic algorithms for the DARP problem, in: *IWINAC* (1), 2007, pp. 498–507.
- [18] W. Herbawi, M. Weber, A genetic and insertion heuristic algorithm for solving the dynamic ride-matching problem with time windows, in: *Proceedings of the Fourteenth International Conference on Genetic and Evolutionary Computation, GECCO '12*, ACM, New York, NY, USA, 2012, pp. 385–392.
- [19] W.E. de Paepe, J.K. Lenstra, J. Sgall, R.A. Sitters, L. Stougie, Computer-aided complexity classification of dial-a-ride problems, *INFORMS J. Comput.* 16 (2) (2004) 120–132.
- [20] A. Coja-Oghlan, S.O. Krumke, T. Nierhoff, A hard dial-a-ride problem that is easy on average, *J. Sched.* 8 (3) (2005) 197–210.
- [21] L. Jun, A genetic algorithm to logistics distribution vehicle routing problem with fuzzy due time, in: *Sixth International Conference on Fuzzy Systems and Knowledge Discovery*, vol. 6, FSKD '09, 2009, pp. 33–39.
- [22] R.R.S. van Lon, T. Holvoet, G. Vanden Berghe, T. Wenseleers, J. Branke, Evolutionary synthesis of multi-agent systems for dynamic dial-a-ride problems, in: *GECCO Companion '12*, ACM, New York, NY, USA, 2012.
- [23] D. Dubois, H. Fargier, P. Fortemps, Fuzzy scheduling: modeling flexible constraints vs. coping with incomplete knowledge, *Eur. J. Oper. Res.* 147 (2003) 231–252.
- [24] S. Kikuchi, Scheduling demand-responsive transportation vehicles using fuzzy-set theory, *Transp. Eng.* 118 (1992) 391–409.
- [25] D. Teodorovic, G. Radivojevic, A fuzzy logic approach to dynamic dial-a-ride problem, *Fuzzy Sets Syst.* 116 (2000) 23–33.
- [26] M. Maalouf, C.A. MacKenzie, S. Radakrishnan, M. Court, A new fuzzy logic approach to capacitated dynamic dial-a-ride problem, in: G. Lim, J.W. Herrmann (Eds.), *Proceedings of the 2012 Industrial and Systems Engineering Research Conference (ISERC)*, Institute of Industrial Engineers (IIE) Annual Conference, Orlando, FL, USA, 2012.
- [27] J.M. Mendel, Fuzzy logic systems for engineering: a tutorial, *IEEE* 83 (3) (1995) 345–377.
- [28] D. Teodorovic, Fuzzy logic systems for transportation engineering: the state of the art, *Transp. Res., Part A* 33 (1999) 337–364.
- [29] D. Teodorovic, K. Vukadinovic, *Traffic Control and Transport Planning: A Fuzzy Sets and Neural Networks Approach*, *Int. Ser. Intell. Technol.*, vol. 13, Springer, 1999.
- [30] A. Kaufmann, M. Gupta, *Introduction to Fuzzy Arithmetic*, Van Nostrand Reinold Company, 1985.
- [31] A. Kaufmann, M. Gupta, *Fuzzy Mathematical Models in Engineering and Management Science*, Elsevier Science, 1988.
- [32] H.-J. Zimmermann, *Fuzzy Set Theory—and Its Applications*, Springer, 2001.
- [33] R.L. Keeney, *Value-Focused Thinking: A Path to Creative Decisionmaking*, Harvard University Press, Cambridge, MA, 1992.
- [34] C.W. Kirkwood, *Strategic Decision Making: Multiobjective Decision Analysis with Spreadsheets*, Duxbury Press, Belmont, 1997.
- [35] E.H. Mamdani, S. Assilian, An experiment in linguistic synthesis with a fuzzy logic controller, *Int. J. Man-Mach. Stud.* 7 (1) (1975) 1–13.
- [36] E.H. Mamdani, Advances in the linguistic synthesis of fuzzy controllers, *Int. J. Man-Mach. Stud.* 8 (1976) 669–678.
- [37] Matlab, *Fuzzy Inference System Modeling: Build Mamdani Systems (GUI)*, MathWorks Documentation Center, 2013, <http://www.mathworks.com/help/fuzzy/building-systems-with-fuzzy-logic-toolbox-software.html>.