Hybrid Genetic Algorithm for an On-Demand First Mile Transit System using Electric Vehicles*

Thilina Perera[1], Alok Prakash[1], Chathura Nagoda Gamage[1]**, and Thambipillai Srikanthan[1]

Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798
pere0004@e.ntu.edu.sg, alok@ntu.edu.sg, chathuratng@gmail.com, astsrikan@ntu.edu.sg

Abstract. First/Last mile gaps are a significant hurdle in large scale adoption of public transit systems. Recently, demand responsive transit systems have emerged as a preferable solution to first/last mile problem. However, existing work requires significant computation time or advance bookings. Hence, we propose a public transit system linking the neighborhoods to a rapid transit node using a fleet of demand responsive electric vehicles, which reacts to passenger demand in real-time. Initially, the system is modeled using an optimal mathematical formulation. Owing to the complexity of the model, we then propose a hybrid genetic algorithm that computes results in real-time with an average accuracy of 98%. Further, results show that the proposed system saves travel time up to 19% compared to the existing transit services.

Keywords: Demand Responsive Transit, Genetic Algorithm, First/Last Mile Problem, Electric Vehicles.

1 Introduction

Public transit systems around the world are constantly challenged to meet the diversified needs of passengers. Not only are these systems expected to provide high quality of service (in terms of reliability and efficiency) but also ensure a high degree of penetration. Furthermore, higher ridership in public transit systems can also lead to easing traffic congestion and pollution in addition to economic gains. Consequently, in the United States, growth of public transit trips outplay both population growth and vehicle miles traveled (VMT) [3].

A public transit journey typically consists of multiple legs and is served by different modalities of transit, such as buses, trains, subways, etc. Thus, in the first leg of a multi-modal public transit journey, a passenger travels to the nearest public transit node, typically a bus stop, to board a bus, which brings him/her to a major transit node. The last leg of the journey is also completed in a similar fashion. However, in a significant number of cases, the nearest public transit node is located outside a comfortable walking distance, typically accepted as 400m [18]. As a result, passengers sometimes have to walk 10-15 minutes to and

* This research project is partially funded by the National Research Foundation Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme with the Technical University of Munich at TUMCREATE.

** Mr. Nagoda Gamage contributed to this research while he was an intern at Nanyang Technological University, Singapore.

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93698-7_8

from the nearest bus stop, mass rapid transit (MRT) or light rail transit (LRT) station in the first and last legs of the journey respectively, which is a significant bottleneck [12]. Thus, the first and last legs are the most troublesome parts of a public transit journey, termed as the first/last mile (FM/LM) problem.

It has been shown in existing work [18] [25] that the root causes of the FM/LM problem are the fixed routes and designated stops of transit buses serving the FM/LM. Further, the problem of excessive walking distance is exacerbated with the addition of waiting times at the bus stops, especially during off-peak hours owing to low frequency fixed schedules of the existing transit buses. At the same time, under-utilization of buses is a significant issue in terms of economic viability for the transit service operator. Thus, it is evident that addressing the FM/LM problem has benefits for both passengers and operators.

Mobility solutions that attempt to tackle the FM/LM problem, strive to extend the penetration of public transit systems by linking FM/LM connections [23]. These solutions are categorized as conventional or innovative mobility solutions. Conventional mobility solutions, also termed as transit oriented development, specifically focus on the creation of compact, walkable, pedestrian oriented, mixed-use communities centered around high quality transit systems [22]. On the contrary, private vehicles and taxis, collectively known as single occupancy vehicles (SOV) are also used to bridge the FM/LM. However, due to the negative impact on traffic congestion and environment, SOVs are highly discouraged in urban cities. On the other hand, technological advancements have led to economical, greener and easier innovative mobility solutions such as bike-sharing [15], casual car-pooling [22], personal mobility devices [30] and ride-sharing [23] being proposed. However, due to the added comfort of using motor vehicles (mini buses, vans) and viability to satisfy diversified user groups, ride-sharing is the preferred solution to the FM/LM problem.

Ride-sharing, also known as dial-a-ride, is a mode of demand responsive transit (DRT) system in which drivers traveling towards a single destination, pick-up and drop-off other passengers traveling towards the same destination or traveling along the same route [26]. Static ride-sharing, where all ride requests are known a-priori has been studied for many years [28]. Traditionally, static DRT, also known as para-transit services, have been used for door-to-door transportation services for the elderly and disabled. It has also been deployed in rural areas and areas of low passenger demand, where operating a fixed-route service is not economically viable. On the other hand, dynamic ride-sharing has gained traction only in recent years due to technological advancements in GPS based tracking, smart phones and wireless communications [5]. Thus, dynamic DRT is arguably one of the most popular and fast evolving new mobility options on the market that can be used to conveniently bridge FM/LM inefficiencies for users by providing quick and easy connections to/from public transit nodes [23].

However, existing work on DRT based public transit systems is limited to one or two ride matches [23]. On the contrary, systems which match multiple passengers to a single vehicle require advanced reservations to account for the significantly high computation time of the algorithms [32] [7]. In a public transit system, however, the ride-matches should occur instantaneously and also, the outcome needs to be communicated to both the passenger and the driver at once. Hence, *there is a need for rapid and scalable solutions for DRT in order for it to be useful in public transit systems for FM/LM problem.*

To this end, we propose a DRT solution consisting of a **homogeneous** fleet of **electric** vehicles with fixed capacity and range (in terms of maximum number of passengers and VMT per vehicle) dispersed in a neighborhood, which responds in **real-time**, to the demand of passengers by picking them from their origin and dropping them off at a predetermined nearest rapid transit node. Moreover, the objective of the proposed solution is to **minimize the total passenger travel time** (waiting time + riding time). In addition, each passenger is guaranteed that the maximum travel time is less than a predetermined value. Also, we consider large instances of passenger requests in real-time. Hence, the proposed solutions require scalable, real-time computation of routes and schedules, which, as discussed in Section 2, cannot be obtained from the existing state-of-the-art techniques. Thus, the **contributions** of this work are: (1) an optimal mixed-integer quadratically constrained programming (MIQCP) model, (2) owing to the complexity of the MIQCP model, we then propose a scalable hybrid genetic algorithm which computes near-optimal results in real-time.

The rest of the paper is organized as follows. In Section 2, we discuss the existing state-of-the-art work on the DRT problem and highlight the limitations. Next, in Section 3, we present the proposed methodology. The results and conclusions of the study are discussed in Section 4 and 5 respectively.

2 Related Work

DRT services are classified as static and dynamic based on the mode of operation. In the static mode all requests are known beforehand, while in the dynamic mode requests are received in real-time. However, it should be noted that dynamic DRTs rarely exist in pure form since a number of requests are often known prior to the planning cycle [4]. Similarly, based on the objective, DRTs are classified as problems which (1) minimize costs subject to full demand satisfaction and side constraints, (2) maximize satisfied demand subject to vehicle availability and side constraints [10]. Irrespective of the classification, the basic DRT problem consists of finding routes and schedules for a fleet of m homogeneous vehicles originating at a single depot, serving the requests of n passengers. However, in literature, many variants of the DRT problem are considered with different features such as heterogeneous vehicles, multiple depots, pickup and delivery etc. Further, it should be noted that literature on DRT problems also consider delivery of goods. However, since the focus of this work is on public transit, our discussion is limited to works that consider DRT for passenger transit.

One of the pioneering work of solving the single vehicle DRT problem was presented in [27]. The author presents a dynamic programming based solution to the capacity constrained single vehicle, many-to-many, immediate-request, static and dynamic DRT problems. Later, the work has been extended by introducing new constraints and solution methods. However, owing to an exact approach, the usefulness of this work is limited to instances of low passenger requests (demand).

DOI: 10.1007/978-3-319-93698-7_8

Heuristic approaches such as parallel tabu-search [16] and neighborhood search [17] focus on instances of high demand. However, in reality DRT problems mostly occur for a fleet of vehicles. Thus, the applicability of the heuristic approaches are limited to specific scenarios.

Similarly, there are many exact and heuristic formulations for the multi-vehicle DRT problem such as column generation [13], branch-and-cut [9], tabu-search [11], sequential insertion heuristic [20], parallel insertion heuristic [14] and genetic algorithms [29] [21] [6]. Even though these methods solve instances with relatively high passenger requests to near-optimality, most of the works consider only the traditional use of DRT services. Thus, either the impact of execution time is minimal or the systems require advanced reservations. As reported in [36], solving the DRT for instances with high passenger requests take about 10 hours. Similarly, the work in [7] requires passengers to book seats 4 hours in advance.

On the contrary, we find limited work that focus specifically on DRT based solutions for the FM/LM problem. Perera et al. propose a scalable local-search heuristic for the FM transit problem [26]. Here, the use of time-window constraints limit the optimization capability which in turn facilitates the local-search. Further, the authors do not consider vehicle range and maximum traveling time constraints. Uchimura et al. propose a 3-tier public transit system connecting neighborhoods with major transit nodes using a DRT system [33]. Here, the authors use a genetic algorithm to solve the DRT problem. They present results for instances with 10 passenger requests (demand) computed within 40 seconds. However, such a system is expected to be used by a large number of passengers and hence, the system needs to be tested with instances of high demand to validate the scalability of the algorithm.

Tsubouchi et al. propose a DRT bus system that responds to real-time transit requests of passengers by computing routes and schedules using a cloud computing platform [32]. The authors claim an algorithm with linear time complexity compared to exponential complexity of the state-of-the-art work. However, they test the algorithm with only 5 vehicles and also the shared ride-ratio is significantly low. A similar effort to bridge the FM/LM gap using DRT is proposed in [34] and further improved in [35]. In this work, authors propose to connect people living in sub-urban areas, through a DRT service to a major urban transit node. Here, they use an insertion heuristic for path planning. However, the system requires advanced (one day prior to departure) passenger reservations.

MIT real-time ride-share research [24] highlights the importance of focusing on large employers and personal choice in clustering passengers. The benefit of focusing on large employers is attributed to the increased match rates that contribute to the economic viability. On the other hand, in the case of "Kutsuplus" in Helsinki, Finland [31], low ridership resulted in the seizure of operations. As stated in [31], the main reason for the empty Kutsuplus buses was the difficulty in matching passengers who are going in the same direction around the same time. Further, Kutsuplus was not designed to serve only the FM/LM connections. As such, our study is focused on providing a DRT based FM/LM solution to localities with low penetration of public transit and high population density.

DOI: 10.1007/978-3-319-93698-7_8

3 Methodology

3.1 System Overview

The proposed system comprises of a homogeneous fleet of electric vehicles (supply) dispersed in a neighborhood. The system responds to passenger requests (demand) in real-time by picking them from their origins and dropping them off at a predetermined nearest rapid transit node. We assume that (1) all vehicles have a fixed capacity and range, (2) passengers request the service using a mobile application. All requests, origins and the real-time traffic data are logged using Google Maps APIs [1]. The proposed algorithm is executed periodically to find an optimum schedule for passengers and the fleet of vehicles. However, using a substantially small periodic value, not only are we able to provide real-time service to the passengers but also reduce the overhead incurred by re-optimizing the routes each time a request is logged. Thus, we model the dynamic DRT problem as a set of periodic static DRT problems. The objective of the problem is to devise a set of routes and schedules in real-time to minimize the total travel time of passengers while satisfying the full demand within the constraints, which include vehicle capacity, range and the maximum travel time per passenger. In this paper, we focus on the case where all vehicles are homogeneous.

3.2 Model Formulation

The problem is modeled using the graph-based structure as proposed in [26], which uses a directed acyclic graph. Here, we summarize the model for clarity. We assume that there are m vehicles and n passengers. Further, irrespective of the number of passengers at the same origin, each passenger is modeled using a node in the graph. Table 1 defines the nodes represented in the graph. Table 2 presents the decision variables in the mathematical formulation, x and s. Finally, Table 3 defines the terms used in the mathematical formulation. The decision variable x_{ijk} is defined for each edge (i, j) and each vehicle V_k, where $i \neq j$, $i \neq m+n+1$ & $j \neq 1, 2, 3, \cdots, m$ and s_{ik} is defined for each node i and each vehicle V_k, where $i \neq 1, 2, 3, \cdots, m$. They are defined in Eq. 1 and 2 respectively. Also, it should be noted that l, r, u and t_{ij} are positive integers and d_{ij} is a non-negative integer. The constraints involved in the model are (1) each passenger is serviced by exactly one vehicle; (2) vehicles with assigned passengers start service from origin of the vehicle and end service at the transit node (node ϑ); (3) number of passengers in each vehicle does not exceed the maximum capacity l; (4) vehicle miles traveled (VMT) of each vehicle does not exceed the range r; and (5) maximum travel time of a passenger is bounded by u.

Table 1: Terminology of Nodes in the Graph

Term	Description	Nodes represented in graph
$\varPhi$	set of all nodes	$1, 2, 3, \cdots, m+n+1$
ν	subset of vehicles	$1, 2, 3, \cdots, m$
ρ	subset of passengers	$m+1, m+2, m+3, \cdots, m+n$
ϑ	transit node	$m+n+1$

Table 2: Decision Variables in the Mathematical Formulation

Decision Variable	Type	Description
x_{ijk}	Binary	Vehicle V_k travels along the edge (i,j) from i to j
s_{ik}	Integer	Time vehicle V_k reaches node i

Table 3: Terminology of the Mathematical Formulation

Term	Description
V_v	v^{th} vehicle in the fleet of v vehicles
P_p	p^{th} passenger in the set of p passengers
$P_{i[a]}$	request time of i^{th} passenger
d_{ij}	travel distance from node i to j
t_{ij}	travel time from node i to j
m	number of vehicles
n	number of passengers
l	maximum capacity of a vehicle
r	maximum range of a vehicle
u	maximum travel time of a passenger

$$x_{ijk} = \begin{cases} 1, & \text{vehicle } V_k \text{ travels from node } i \text{ to node } j, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

$$s_{ik} = \begin{cases} f, & \text{vehicle } V_k \text{ services node } i, \ f \epsilon z^+, \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

3.3 Optimal Mathematical Formulation

Based on the model in Section 3.2, we present an optimal mixed-integer quadratically constrained programing (MIQCP) mathematical formulation. Equation 3 shows the objective function of the mathematical formulation, which minimizes the total travel time of all passengers. All constraints in the model given in Eq. 5 - 18 are classified into routing, timing, side and subtour elimination, which are explained in detail subsequently.

Objective function:

$$minimize \sum_{i \in \rho} \sum_{j \in \Phi} \sum_{k \in \nu} (x_{ijk} * s_{\vartheta k} - P_{i[a]}); \tag{3}$$

Subject to:

$*$ *Routing Constraints*

$$\sum_{j \in \rho} x_{kjs} = 0 \quad \forall k \in \nu, \ \forall s \in \nu; \quad where \quad k \neq s; \tag{4}$$

$$\sum_{j \in \rho} x_{kjk} \leq 1 \quad \forall k \in \nu; \tag{5}$$

$$x_{\vartheta jk} = 0 \quad \forall j \in \Phi, \ \forall k \in \nu; \tag{6}$$

$$x_{ikk} = 0 \quad \forall i \in \rho, \ \forall k \in \nu; \tag{7}$$

DOI: 10.1007/978-3-319-93698-7_8

$$\sum_{i \in \Phi} x_{ibk} - \sum_{j \in \Phi} x_{bjk} = 0 \quad \forall k \in \nu, \forall b \in \rho; \tag{8}$$

$$\sum_{i \in \rho} x_{i\vartheta k} \leq 1 \quad \forall k \in \nu; \tag{9}$$

$$\sum_{i \in \rho} \sum_{k \in \nu} x_{i\vartheta k} \geq 1 \quad \forall k \in \nu; \tag{10}$$

∗ *Timing Constraints*

$$s_{kk} = 0 \quad \forall k \in \nu; \tag{11}$$

$$x_{ijk}(s_{ik} + t_{ij} - s_{jk}) \leq 0 \quad \forall i \in (\Phi \backslash \vartheta), \forall j \in \Phi, \forall k \in \nu; \tag{12}$$

∗ *Side Constraints*

$$\sum_{i \in \rho} \sum_{j \in \Phi} x_{ijk} \leq l \quad \forall k \in \nu; \tag{13}$$

$$\sum_{i \in \Phi} \sum_{j \in \Phi} x_{ijk} * d_{ij} \leq r \quad \forall k \in \nu; \tag{14}$$

$$s_{\vartheta k} \leq u_l \quad \forall k \in \nu;\ u_l = min(l_{m+1}, l_{m+2}, \cdots, l_{m+n}); \tag{15}$$

∗ *Subtour Elimination Constraints*

$$x_{k\vartheta s} = 0 \quad \forall k \in \nu,\ \forall s \in \nu; \tag{16}$$

$$x_{iik} = 0 \quad \forall i \in \rho,\ \forall k \in \nu; \tag{17}$$

$$\sum_{j \in \Phi} \sum_{k \in \nu} x_{ijk} = 1 \quad \forall i \in \rho; \tag{18}$$

The first routing constraint in Eq. 4 sets the origin of the routes to each vehicle location. Equation 5 ensures that at least one vehicle moves from its' origin to serve passenger requests. Equation 6 and 7 prevents routes from originating at the transit node and finishing at vehicle nodes respectively. Equation 8 ensures that a route does not end at a passenger node. The final two routing constraints deal with the completion of the route. Equation 9 ensures that a vehicle will reach the transit node only through a passenger node. Similarly, Eq. 10 enforces that at least one vehicle reaches the transit node. Thus, in the best case one vehicle serves the passenger requests and reaches the transit node.

Timing constraints in Eq. 11 and 12 initializes the service time of each vehicle and models the travel time between nodes respectively. Side constraints, namely vehicle capacity, range and maximum travel time of passengers are modeled in Eq. 13, 14 and 15 respectively. Also, it should be noted that, we assume the range of the vehicle is directly comparable to the VMT. Thus, in Eq. 14 the summation of the VMT is compared with the maximum allowable range. In Eq. 15 we assume that the upper bound of the maximum riding time for each vehicle is equal to the minimum value among all passengers, irrespective of the assigned passengers. This assumption is realistic since the minute periodicity of

DOI: 10.1007/978-3-319-93698-7_8

the algorithm prevents significant deviations in the travel time among the passengers. Subtour elimination constraints further tighten the model and prevents any loops or subtours. Equation 16 prevents the vehicles traveling directly from origin to the transit node. Equation 17 prevents loops in the route. Finally, Eq. 18 ensures that there is only one outgoing edge at a passenger node.

The solution from the above formulation provides an optimal set of routes and schedules for the fleet of vehicles. However, due to the complexity of the model (NP-hard) [14], execution time grows exponentially with the size of the problem . Thus, we are motivated to develop a scalable approach that can solve the problem in real-time with significant accuracy for large instances.

3.4 Hybrid Genetic Algorithm

Genetic algorithms (GA) use principles of natural evolution to model real-world problems. Generally, they produce high quality results for complex combinatorial optimization problems [19]. In the past, GAs have been successfully used in various VRPs [29] [21] [6]. GAs can be easily adapted and improved for different problems using knowledge local to the problem. Further, they can be easily controlled to complete within a predefined number of iterations which favors real-time computations. Thus, we propose a hybrid meta-heuristic combining a modified GA with local-search and savings heuristic [8]. The pseudo-code of the proposed hybrid genetic algorithm (HGA) is given in Algorithm 1.

Algorithm 1 Pseudo Code of the Hybrid Genetic Algorithm

```
Input: passenger requests ($P$), fleet of vehicles ($V$)
Output: route & schedule of the fleet of vehicles
INITIALIZE population
FITNESS EVALUATION of each candidate

repeat(
    i: PARENT SELECTION;
    ii: MUTATION;
    exchange mutation
    insertion mutation
    iii: PARENT SELECTION;
    iv: CROSSOVER;
    modified heuristic crossover
    re-insertion crossover
    )
until TERMINATION CONDITION is satisfied
```

Genetic Encoding and Fitness Calculation: We use **path based representation**, to depict a single chromosome. Each chromosome consists of $m * l$ genes, where m and l are defined in Table 3. Thus, we effectively eliminate infeasible solutions in terms of capacity. Further, we use **integer encoding** to represent each passenger. The same objective function given in Eq. 3 is used as the fitness function. Thus, for each vehicle the fitness function evaluates total travel time of all passengers in a vehicle.

Initial Population Construction: The intuitive method of selecting the initial populations in GA is random selection. However, good initial populations can significantly improve the execution time of the algorithm [36]. Thus, we propose to use the method given in [26], which uses a **local-search heuristic** to generate initial solutions. Further, we modify the method in [26] and generate multiple chromosomes along with several random chromosomes.

Parent Selection: Based on the encoding scheme, fitness value of each gene is directly proportional to the fitness value of the chromosome. Thus, we use the **roulette wheel selection** procedure for parent selection. However, as the objective of the work is minimization of total travel time, chromosomes with higher fitness values (weak parents) are considered as candidates for reproduction. Further, in order to maintain strong parents across generations, the two most **elite** parents are directly included in the next generation.

Mutation: Mutation is performed on a single individual and it helps to explore new states and avoid local optima. In our algorithm, we use two mutation operators, namely **exchange** and **insertion** mutation. In the exchange mutation operator, a selected sub-tour is swapped across a randomly selected point. However, in the proposed HGA a **local optimization** is used to find a suitable point. In the insertion mutation, a gene is removed from the tour and inserted back into a different position in the tour. However, when inserting we use the **savings heuristic**.

Crossover: Crossover operator improves the average quality of the population by simulating the reproduction between two individuals. As in mutation, we propose two crossover operators. The first crossover operator is a **modified heuristic crossover**, where genes with the highest fitness value of the selected parents are swapped to produce two new individuals. In the next crossover operator the gene with the highest fitness value is removed from the selected parent and inserted into the path of another randomly selected parent. The re-insertion is done based on the **savings heuristic**. Thus, the fitness value of the parent with the removed gene improves in contrast to the other parent. However, as the goal is collectively reducing the fitness value this approach may yield better results at the expense of one parent.

4 Results

In this section, we present the results of the computational experiments to verify the proposed HGA. *IBM ILOG CPLEX OPTIMIZATION STUDIO 12.7.1* [2] is used to implement the mathematical model in Section 3.3 and the in-built constraint programming solver is used to solve the formulation to optimality. The HGA proposed in Section 3.4 is implemented in C++. Runtime measurement is done on a PC with 32 GB RAM, running Windows 10 on an Intel Xeon E5-1630V3 CPU at 3.70 GHz.

4.1 Experimental Setup

The motivation of our study is to devise an algorithm capable of handling large instances of passenger requests in real-time. This is a common scenario within the premises of any university, thus we select a university for our experimental

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93698-7_8

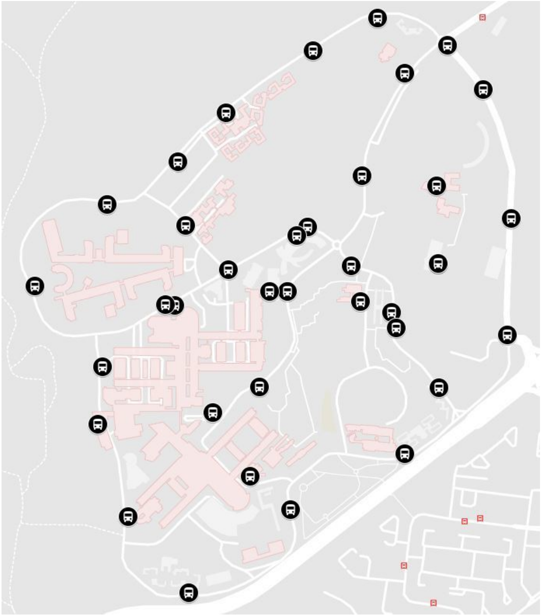

Fig. 1: Sample Bus Stop Distribution

setup shown in Fig. 1. Here, demand in terms of passenger requests can originate from any location within the given zone. However, for clarity we limit the origins of demand to a set of fixed locations (bus stops), indicated in Fig. 1. In contrast, vehicles (supply) are considered to be dispersed in the zone. Also, it is noteworthy that there may be multiple passengers in a single origin for a given problem.

4.2 Experiments

Given the complexity of the optimal mathematical formulation and hence, the exponential increase in time required to obtain results, we have divided our experiments into two categories. First, we study 10 small-sized problems, where we compare the accuracy of the proposed Hybrid Genetic Algorithm (HGA) with the optimal results. The next set of experiments comprise of 30 medium-large problems. However, in this case we compare the results of the proposed HGA with two realistic upper and lower bounds obtained from Google Maps APIs [1]. Also, each problem is repeated 5 times and for each instance the origins of demand and supply are selected randomly. Further, in each instance passengers are distributed randomly among the origins. In both experiments, the maximum capacity and driving range of a vehicle are set to constant values of 8 and 30 km respectively. Also, we ensure that the supply (available seats) is equal to or greater than the demand (number of passengers).

4.3 Evaluation

Evaluation of the algorithm is performed under three criteria. Criteria 1: Scalability of the proposed HGA, Criteria 2: Comparison of the proposed HGA to the optimal results and bounds obtained through *CPLEX* and Google Maps respectively, and Criteria 3: Analysis of the travel time saving with vehicle utilization.

DOI: 10.1007/978-3-319-93698-7_8

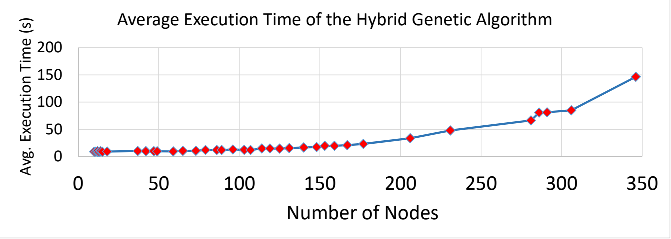

Fig. 2: Scalability of the Hybrid Genetic Algorithm

Scalability: As mentioned in Section 3.3, the time complexity of the optimal mathematical formulation is NP-hard. However, the proposed HGA computes the routes and schedules in a significantly low time. The average execution time of the 5 instances of all the 40 problems are shown Fig. 2. Here, the horizontal and vertical axes represent the number of nodes $(m + n + 1)$ and the average execution time of the algorithm in seconds respectively. Here, we observe the significantly low increase of average execution time with the problem size.

Performance: Performance of the proposed HGA, in terms of accuracy of results compared to the optimal values (experiment 1) and bounds (experiment 2) are analyzed. In experiment 1, we compare the results of *CPLEX* and the proposed HGA. However, for experiment 2 we define two bounds obtained from Google Maps for each problem. For the **lower bound (LB)**, we assume that all passengers use their own vehicles to travel to the transit node (SOV time). In contrast, for the **upper bound (UB)** we obtain the time taken for the journey using public transit (Transit time). However, when calculating the UB, we assume that buses have sufficient capacity to cater the demand and also use schedules of off-peak day time. Results and the corresponding parameters for the two experiments are given in Table 4 and 5 respectively. In both tables, we present the average total travel time for the passengers in minutes. In experiment 1, the average deviation of results from the optimal values is **1.72%**. In experiment 2, 28 problems are within the bounds. However, in problem 13 and 19 the HGA result is marginally higher than the UB. The reasons for the deviation is analyzed in the subsequent section. Further, due to the randomness of passenger and vehicle allocation we observe an outlier in problem 3.

Variation of Travel Time Savings: Travel time saving against the vehicle utilization is shown in Fig. 3. Here, the horizontal and vertical axes represent vehicle utilization and travel time saving percentage respectively. Also, the corresponding problem number is marked in the graph. Travel time saving implies the benefit passengers' gain in terms of travel time by using the proposed DRT system with respect to public transit. In general, when the vehicle utilization is moderate, the proposed system outperforms the transit system with savings of up to **19%**. However, when vehicle utilization is significantly high ($\geq 95\%$) the benefit obtained from the proposed system is marginal. These points lie on the bottom right of the graph. Hence, problems 13 and 19 with vehicle utilization of 98% and 97% respectively under-perform compared to the transit system.

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93698-7_8

Table 4: Results for Experiment 1

Problem	Parameters			No of	HGA	CPLEX	Deviation
No	Vehicles	Passengers	Bus Stops	Nodes	(min)	(min)	(%)
1	4	5	5	10	72.2	71.6	0.83
2	4	6	6	11	95.6	93.6	2.17
3	4	7	7	12	106.4	104.4	1.99
4	4	8	8	13	131	126.2	3.73
5	5	6	6	12	84.8	84.4	0.47
6	5	7	7	13	102	101	0.98
7	5	8	8	14	117.4	115.2	1.95
8	6	7	7	14	95.8	95.8	0.00
9	6	8	8	15	117	114.6	2.10
10	7	10	10	18	143.8	139.6	3.00

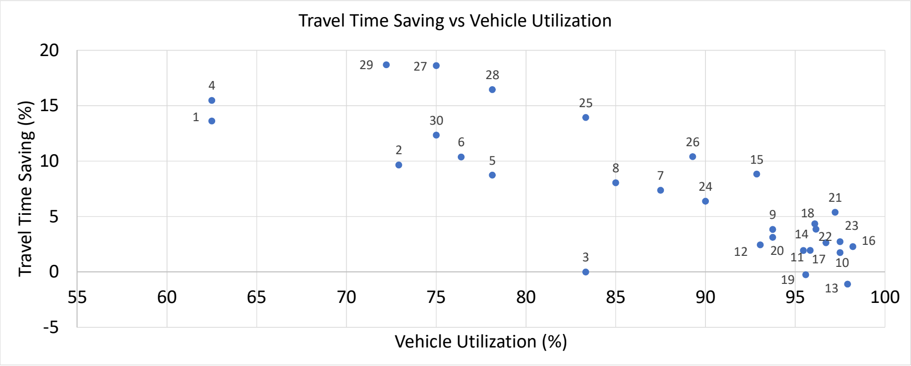

Fig. 3: Travel Time Saving of the Proposed DRT System

5 Conclusion

This paper proposes a scalable hybrid genetic algorithm (HGA) for an on-demand first mile transit system using electric vehicles. The problem is first modeled using an optimal mathematical formulation and solved for small instances. Next, we propose a HGA, which computes near-optimal results in real-time. Further, the proposed system achieves considerable travel time savings with respect to public transit services. In future, we plan to extend the work for heterogeneous vehicles (in terms of capacity and range) and for last mile transit.

References

1. Google Maps APIs: Build the next generation of location experiences. `https://developers.google.com/maps/`
2. IBM ILOG CPLEX Optimization Studio. `https://ibm.co/2vXgzRC`
3. Association, A.P.T.: Public Transportation Benefits. `http://www.apta.com/mediacenter/ptbenefits/Pages/default.aspx` (2017)
4. Borndörfer, R., et al.: Telebus Berlin: Vehicle Scheduling in a Dial-a-Ride System (1999)
5. Cao, B., et al.: Sharek: A scalable dynamic ride sharing system. In: MDM (2015)

ICCS Camera Ready Version 2018
To cite this paper please use the final published version:
DOI: 10.1007/978-3-319-93698-7_8

Table 5: Results for Experiment 2

Problem	Parameters			No of	HGA	LB	UB
No	Vehicles	Passengers	Bus Stops	Nodes	(min)	(min)	(min)
1	6	30	30	37	652.4	325.6	755.2
2	6	35	35	42	804.2	382.6	890
3	6	40	40	47	1014.2	435	1014
4	8	40	25	49	848.8	438.6	1004.2
5	8	50	30	59	1152.4	539.2	1262.6
6	9	55	32	65	1236.8	606.6	1379.8
7	9	63	34	73	1487.8	682.4	1606
8	10	68	25	79	1576	738.6	1713.8
9	10	75	30	86	1825.2	810.8	1897.8
10	10	78	18	89	1952.8	841	1987.2
11	11	84	27	96	2062	900.6	2102.4
12	12	90	30	103	2183.2	986	2253.4
13	12	94	21	107	2386	1024	2359.8
14	13	100	22	114	2393.4	1078.2	2489.2
15	14	104	25	119	2476.6	1110	2716.2
16	14	110	40	125	2764	1188.6	2828.4
17	15	115	35	131	2831.8	1258.6	2887.8
18	16	123	19	140	2965.2	1341.2	3099.4
19	17	130	28	148	3187	1383.6	3178.8
20	18	134	37	153	3241.4	1464	3322.4
21	18	140	26	159	3431.4	1492.6	3626.2
22	19	147	17	167	3568.6	1599.8	3664.8
23	20	156	23	177	3816.8	1700.2	3923.4
24	25	180	27	206	4207.8	1962	4494
25	30	200	32	231	4448.6	2172.8	5168.4
26	35	250	16	286	5694.2	2703.8	6355.2
27	40	240	26	281	5081.4	2591.8	6243.4
28	40	250	25	291	5267.8	2678.8	6304.8
29	45	260	37	306	5422.6	2805	6668.8
30	45	300	25	346	6617.2	3214.6	7548.8

6. Carballedo, R., et al.: A New Evolutionary Hybrid Algorithm to Solve Demand Responsive Transportation Problems (2011)
7. Chevrier, R., et al.: Comparison of three algorithms for solving the convergent demand responsive transportation problem. In: ITSC (2006)
8. Clarke, G., Wright, J.W.: Scheduling of vehicles from a central depot to a number of delivery points. Operations Research **12**(4), 568–581 (1964)
9. Cordeau, J.F.: A branch-and-cut algorithm for the dial-a-ride problem. Operations Research **54**(3), 573–586 (2006)
10. Cordeau, J.F., Laporte, G.: The dial-a-ride problem (darp): Variants, modeling issues and algorithms. 4OR (2003)
11. Cordeau, J., Laporte, G.: A tabu search heuristic for the static multi-vehicle dial-a-ride problem. Transp Res Part B: Methodological **37**(6), 579 – 594 (2003)
12. CREATE, T.: The last mile problem. `http://bit.ly/2qKpCaY` (2012)
13. Desrosiers, J., et al.: A dynamic programming solution of the large-scale single-vehicle dial-a-ride problem with time windows. Amer. J. Math. Management Sci

6(3-4), 301–325 (1986)
14. Diana, M., Dessouky, M.M.: A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows. Transp Res Part B: Methodological **38**(6), 539 – 557 (2004)
15. Elliot, F.: Bikeshare: A review of recent literature. Transp Rev **36**(1), 92–113 (2016)
16. Gendreau, M., et al.: A dynamic model and parallel tabu search heuristic for real-time ambulance relocation. Parallel Computing **27**, 1641–1653 (2001)
17. Gendreau, M., et al.: Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries. Transp Res Part C Emerg Technol **14**(3), 157 – 174 (2006)
18. Gibson, A.: Eliminating Public Transits First-Mile/Last-Mile Problem. `http://bit.ly/2mgw6sm` (2016)
19. Gonzlez, M., et al.: Using genetic algorithms for maximizing technical efficiency in data envelopment analysis. Procedia Comput. Sci. **51**(C), 374–383 (Sep 2015)
20. Jaw, J.J., et al.: A heuristic algorithm for the multi-vehicle advance request dial-a-ride problem with time windows. Transp Res Part B: Methodological **20**(3), 243 – 257 (1986)
21. Jorgensen, R.M., et al.: Solving the dial-a-ride problem using genetic algorithms. J Oper Res Soc **58**(10), 1321–1331 (Oct 2007)
22. Lesh, M.C.: Innovative concepts in first-last mile connections to public transportation. In: Int. Conference on Urban Public Transportation Systems (2013)
23. Logan, S., et al.: First and Last Mile Connections: New Mobility. `http://on.nrdc.org/2utPxUd` (2016)
24. MIT: Welcome to MIT Real-Time Rideshare Research. `http://ridesharechoices.scripts.mit.edu/home/` (2010)
25. Osaba, E., et al.: An asymmetric multiple traveling salesman problem with backhauls to solve a dial-a-ride problem. In: SAMI (2015)
26. Perera, T., et al.: A scalable heuristic algorithm for demand responsive transportation for first mile transit. In: INES (2017)
27. Psaraftis, H.N.: A dynamic programming solution to the single vehicle many-to-many immediate request dial-a-ride problem. Transp. Sci **14**(2), 130–154 (1980)
28. Psaraftis, H.N., et al.: Dynamic vehicle routing problems: Three decades and counting. Networks **67**(1), 3–31 (2016)
29. Rekiek, B., et al.: Handicapped person transportation: An application of the grouping genetic algorithm. Eng Appl Artif Intell **19**(5), 511 – 520 (2006)
30. Robyn, D., et al.: Use of personal mobility devices for first-and-last mile travel: The macquarie ryde trial. In: ARSC (2015)
31. Sulopuiso, O.: Why Helsinki's innovative on-demand bus service failed. `http://bit.ly/2mkeUDg` (2016)
32. Tsubouchi, K., et al.: Innovative on-demand bus system in japan. IET Intelligent Transport Systems (2010)
33. Uchimura, K., et al.: Demand responsive services in hierarchical public transportation system. IEEE Transactions on Vehicular Technology (2002)
34. Uehara, K., et al.: A proposal of a transport system connecting demand responsive bus with mass transit. In: ICCE (2014)
35. Uehara, K., et al.: Evaluation of a hierarchical cooperative transport system using demand responsive bus on a dynamic simulation. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences (2016)
36. Xiang, Z., et al.: A fast heuristic for solving a large-scale static dial-a-ride problem under complex constraints. Eur J Oper Res **174**(2), 1117 – 1139 (2006)

DOI: 10.1007/978-3-319-93698-7_8