ISSN 23499842(Online), Volume 1, Special Issue 2(ICITET 15), March 2015

International Journal of Innovative Trends and Emerging Technologies

PROBLEMS ARRANGEMENTS USING GENETIC ALGORITHMS SOLUTION METHOD

D.SANGEETHA[1] D.SHOBANA[2], K.SOWMIA[3] O.SANDHYA(Asst.Prof)
Department of Computer Engineering ,
ADHIPARASAKTHI ENGINEERING COLLEGE

Abstract—

carpoolers the use of the carpool services via a smart handheld device anywhere and at any time. The carpool service agency in the ICS is integrated with the abundant geographical, traffic, and societal information and used to manage requests. For help in coordinating the ride matches via the carpool service agency, we apply the genetic algorithm to propose the genetic-based carpool route and matching algorithm (GCRMA) for this multi objective optimization problem called the carpool service problem (CSP). The experimental section shows that the proposed GCRMA is compared with two single-point methods: the random-assignment hill climbing algorithm and the greedy-assignment hill climbing algorithm on real-world scenarios. Use of the GCRMA was proved to result in superior results involving the optimization objectives of CSP than other algorithms. Furthermore, our GCRMA operates with significantly a small amount of computational complexity to response the match results in the reasonable time, and the processing time is further reduced by the termination criteria of early stop.

Keywords- Carpool service problem (CSP), genetic algorithm, Advanced intelligent carpool system (AICS). Advanced genetic-based carpool route and matching algorithm (AGCRMA)

I. INTRODUCTION

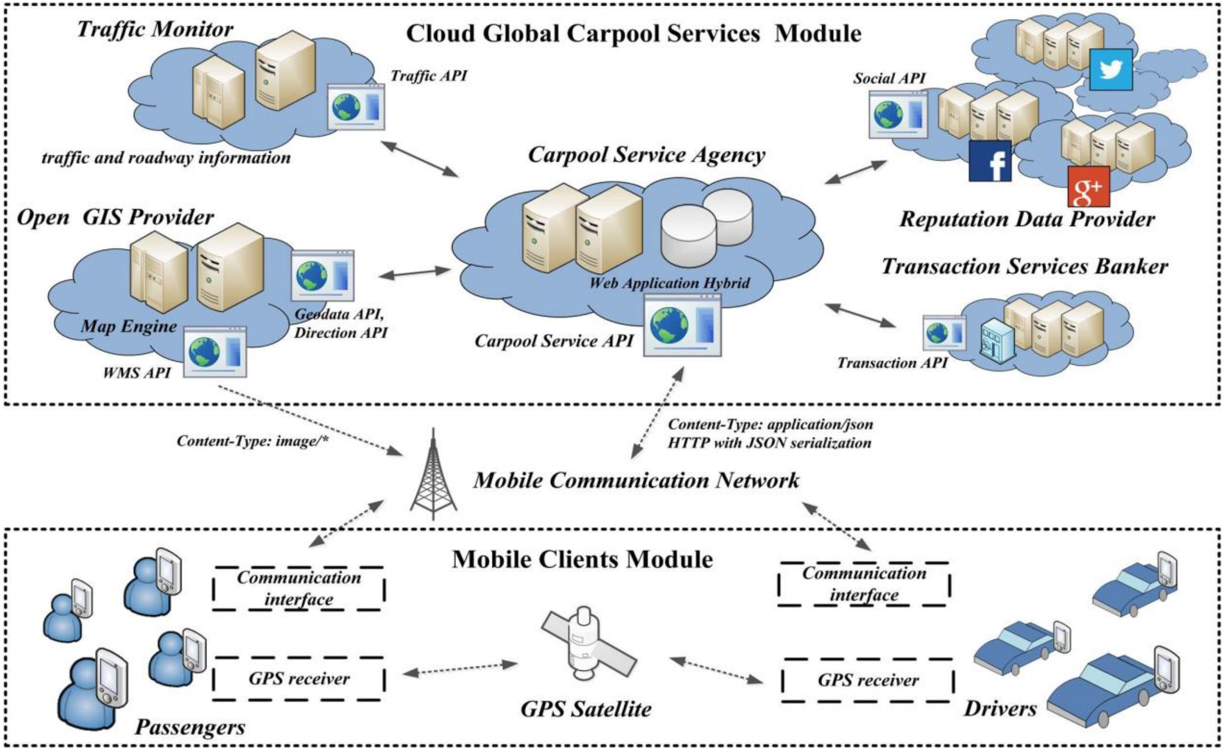
The rising car usage deriving from growth in jobs and residential population causes air pollution, energy waste and consumption of people's time. Public transport cannot be the only answer to this increasing transport demand. Carpooling, which is based on the idea that sets of car owners having the same travel destination share their vehicles has emerged to be a viable possibility for reducing private car usage in congested areas [1]. Its actual practice requires a suitable information system support and, the most important, the capability of effectively solving the underlying combinatorial optimization problem. Genetic algorithms (GAs) have seen widespread used amongst modern meta-heuristics, however, GAs do not appear to have made a great impact so far on the carpooling problem. The aim of this study is to put forward a conceptually straightforward GA for carpooling problem, which is competitive with other modern heuristics in terms of computing time and solution quality. Carpooling (also car-sharing, ride-sharing, lift-sharing and covoiturage), is the sharing of car journeys so that more than one person travels in a car. By having more people

using one vehicle, carpooling reduces each person's travel costs such as fuel costs, tolls, and the stress of driving. Carpooling is seen as a more environmentally friendly and sustainable way to travel as sharing journeys reduces carbon emissions, traffic congestion on the roads, and the need for parking spaces. Authorities often encourage carpooling, especially during high pollution periods and high fuel prices.

Carpooling is a relatively environmentally sound system of transportation in which empty seats are offered to additional passengers and has been found to be one of the best solutions to traffic congestion [5], [6]. Drivers share their cars with one or more people who have similar transportation routes. By reducing the number of empty seats in these vehicles, occupancy rates are significantly increased.

Consequently, fewer vehicles would be required to transport the same quantity of commuters to their respective destinations, resulting in substantially fewer cars on the road. Other carpooling benefits include reductions in travel cost, energy consumption, and vehicle emissions. As a result of technological advances such as the development of smart handheld device software and hardware, along with mobile Internet technology, the website-based carpool system has become more advanced and is now appropriately referred to as the intelligent carpool system (AICS). The chromosome is encoded as a serial of pools; each pool is expressed by sequential numbers where each number indicates a client in this pool. We use a 2-point crossover operator to obtain offspring solutions, parts of genes from each parent are selected and recombined. Then the duplicate client caused by the recombination will be removed, and the client who does not exist in any part of genes will be inserted into the chromosome based on the preference matrix. For client i, his preference to be pooled in carpool h (preferenceih) is calculated as the sum of the preference value of client i to all the clients who already have been pooled in carpool h. The choice of a carpool is performed by a roulette wheel selection principle, the probability of client i will be inserted into carpool h is computed using Eq.

Our paper focuses on solving the problem configured with the Single driver to many passenger arrangements. The carpool service problem (CSP) is an NP-hard problem, where the location of each participant including the driver and the passenger is a complicated origin–destination pair [14], [15]. The authors in [14] proposed a method using the application of integer programming to solve the carpool problem in a workplace environment, by which to facilitate the sharing of employee vehicles. Nevertheless, integer programming, which belongs to the family of exact optimization, is a deterministic method that always obtains the same approximate solutions in different runs. As such, it is not very effective at generating adequate solutions for a large number of carpool users.



Moreover, an advanced method presented by others in [15] uses the genetic algorithm in which the design of recombination is simply implemented via a single-point crossover operator and the mutation procedure uses five basic operators: 1) a push backward operator; 2) a push forward operator; 3) a remove–insert operator to randomly select an origin–destination pair for removal from the route; 4) a transfer operator to pick an origin–destination pair from the route for insertion into other routes; and 5) a swap mutation operator to swap a random point and a neighbor point en route. Unfortunately, these operators are neither effective nor productive because all perform in a random manner that lacks a problem specific orientation necessary to tailoring to the characteristics of the carpool problem. Moreover, the genetic algorithm has been successfully applied in many situations [16]–[19] and is able to determine solutions with near-optimal quality within a reasonable amount of time. In this paper, we present a movement model by which to characterize the moving patterns of residents living throughout the metropolitan area. In [29] proposed a genetic-based carpool route and matching algorithm (AGCRMA) with which to solve the CSP by dramatically acquiring optimal match solutions while reducing the required computing time.

In this paper we proposed a advanced genetic-based carpool route and matching algorithm (AGCRMA) with which to solve the CSP by acquiring optimal match solutions while reducing the required computing time with use of genetic algorithms instead of mathematical calculations only.

The rest of this paper is organized into four more sections. Section II describes the applied framework of the ICS and defines the CSP. Current GCRMA is described in Section III. Section IV we presents proposed AGCRMA. Finally, our conclusions are drawn in Section V.

II. CURRENT SYSTEM MODEL

A. AICS

A cloud-computing-based carpool services framework called the AICS. The AICS system is built by the structure of the web application hybrid and is based on service orientation [20]. It comprises two primary modules: a mobile clients (MC) module and a cloud global carpool services (CGCS) module. Communication can be established between the MC module and the CGCS module by using the web HTTP protocol via the mobile communication network.

1) A. MC Module

In order to give system users the opportunity to obtain carpool matches anywhere and at any time, drivers and passengers alike can use the MC module to perform carpool operations (e.g., requesting and offering rides) via their mobile devices. The MC module is a mobile application built on an advanced mobile operating system such as iOS, Android, Windows Phone, and so on. It features an integrated GPS receiver and capability for mobile communication. Because of this, users can obtain information about their current locations by automatically accessing the GPS signals of satellites and can also retrieve geo resource map images over the Web Map Service (WMS) application programming interface (API) to precisely pinpoint their pickup and destination locations. Using the MC module, users can both offer carpool rides as drivers and send carpool requests as passengers. When drivers and passengers are in the same regional range, a series of users' offers and requests will help them find suitable carpool partners. Drivers can pick up assigned passengers from their departure locations and drop them off at destination locations if the carpool matches are received and allowed by users. Completion notification is sent to each user after each ride, and both the driver and the passenger have the option of rating their respective experiences with each other, which are viewable to future potential carpool partners.

2) CGCS Module

To enable global implementation of the AICS on cross platform devices, the CGCS module provides the Restful web service application interface to support interoperability with the MC module. The CGCS uses mashup development to seamlessly aggregate a great number of powerful web service applications, including

1) An OpenGIS provider, 2) a traffic monitor, 3) a reputation data provider, and 4) a transaction services banker, to enrich the functions and capabilities of the AICS system. The OpenGIS provider facilitates the usability of the MC module through the published WMS API. Moreover, the OpenGIS provider is used to supply digital road information regarding complicated features such as one-way streets, two-way streets, and so on. It actively contributes to routing functionality when one of the matching optimization algorithms is applied to the system. Traffic status is ordinarily monitored by a department within the government. For consideration of real-time traffic factors, the CGCS module integrates OpenGIS content with the traffic API published by the traffic monitor, thereby reorganizing traffic information and enhancing route and match estimations. Many well-known social platforms store types of social information such as connections, comments, recommendations, and so on, all of which can be securely accessed through the open authorization standard OAuth [21], which authorizes third party access to these resources. These social terms and ratings of each user are systematically normalized as a credit score, by which to establish interpersonal trust and responsibility in the carpool system.

3) Definition of the CSP

Under the service-oriented carpool system in cloud computing, the route and matching mechanics of the CSP is a multi objective optimization problem in that it simultaneously concerns more than one objective. Drivers travel to all origin–destination pairings (pickup and drop-off locations) of assigned passengers who have submitted carpool request to the service agency of the AICS system.

The primary objective of the CSP is to maximize the total number of passengers matched with drivers (NM), as well as their cumulative credit scores (TC). The secondary objective is to minimize the average travel distance of drivers (AD), the average waiting distance of passengers (AWD), and the average travel distance of passengers (ATD).

III. PROPOSED SYSTEM (AGCRMA)

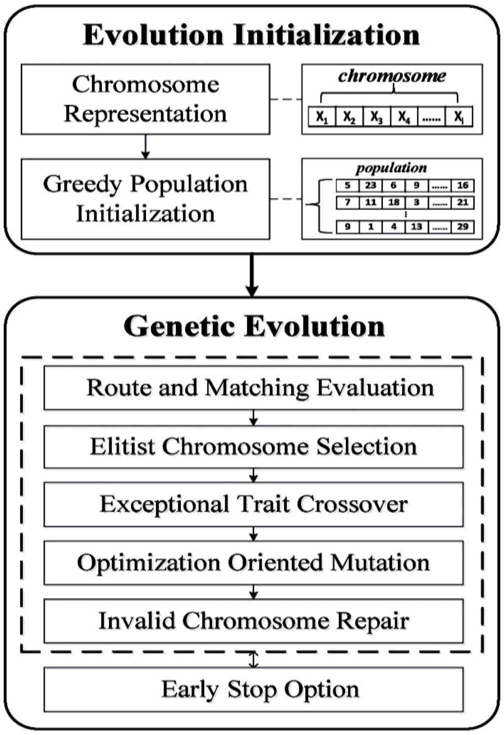


Fig.2 Proposed AGCRMA

In This Paper we describe our proposed AGCRMA, which determines carpool matches via the ICS and thereby provides an effective solution to the CSP. As shown in Fig. 2, the AGCRMA consists of two important modules: 1) an evolution initialization (EI) module and 2) a genetic evolution (GE) module. The EI module initializes chromosomes using chromosome representation and greedy population initialization to effectively generate initial solutions to the CSP problem. Carpool requirement properties are expressed by the EI module through the chromosome representation procedure that encodes CSP solutions into chromosomes according to driver and passenger requirements. Those candidates of the initial population pool that are determined to be feasible matches are then generated by the greedy population initialization procedure via distance-based heuristics. Upon generation of effective initial solutions in the EI module, the proposed GE module accurately determines optimum solutions to the CSP by heuristically simulating natural evolution via six proposed procedures: First, the route and matching evaluation takes place, after which elitist chromosome selection occurs. Then, exceptional trait crossover is implemented, followed by optimization-oriented mutation and invalid chromosome repair. Finally, the early stop option is employed. Repetition of this procedure many times will yield approximate optimal solutions to the CSP.

1) EI Module

1) Chromosome Representation:

Each carpool request (Req.) is represented by a quintet (RID, RN, RL, RD, RC) that contains several properties, such as the identity number RID, the seat number RN, the current location RL, the destination RD, and the request user category

RC: 1 if submitted by a driver.

RC: 0 if submitted by a passenger.

Before the algorithmic process begins, each input request appends an additional field of credit score RS as a sextet (RID, RN, RL, RD, RC, RS), which is input into the proposed AGCRMA. Using these, the exact requests of driver and passenger can be denoted as

Req(RID,RN,RL,RD,RS) =Di, if RC = 1 The carpool request includes *m* drivers and *n* passengers, which can be expressed as

Req(RID,RN,RL,RD,RS) =Pj , if RC = 0.

The carpool request includes m drivers and n passengers, which can be expressed as D_{mx5} and P_{nx5} respectively.

$$D_{m\times 5} = \begin{bmatrix} D_1 \\ D_2 \\ \vdots \\ D_m \end{bmatrix} = \begin{bmatrix} d_1^{ID} & d_1^{N} & d_1^{L} & d_1^{D} & d_1^{S} \\ d_2^{ID} & d_2^{N} & d_2^{L} & d_2^{D} & d_2^{S} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ d_m^{ID} & d_m^{N} & d_m^{L} & d_m^{D} & d_m^{S} \end{bmatrix}$$

$$P_{n\times 5} = \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_n \end{bmatrix} = \begin{bmatrix} p_1^{ID} & p_1^{N} & p_1^{L} & p_1^{D} & p_1^{S} \\ p_2^{ID} & p_2^{N} & p_2^{L} & p_2^{D} & p_2^{S} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ p_n^{ID} & p_n^{N} & p_n^{L} & p_n^{D} & p_n^{S} \end{bmatrix}.$$

Effective construction of an adaptive chromosome that can flexibly express properties of the carpool requests is very important. To accomplish this, we propose a two-layer representational procedure, which is composed of 1) an assignment layer and 2) an implicit routing layer. This assignment layer can be divided into several segments, with each segment further subdivided into genes. The index of each passenger is represented by the value of each gene. The passenger indices within the genes are grouped and then assigned to a driver for each segment. As such, this design can effectively represent the number of drivers to whom passengers are allocated.

Effective construction of an adaptive chromosome that can flexibly express properties of the carpool requests is very important. To accomplish this, we propose a two-layer representational procedure, which is composed of 1) an assignment layer and 2) an implicit routing layer. This assignment layer can be divided into several segments, with each segment further subdivided into genes. The index of each passenger is represented by the value of each gene. The passenger indices within the genes are grouped and then assigned to a driver for each segment. As such, this design can effectively represent the number of drivers to whom passengers are allocated.

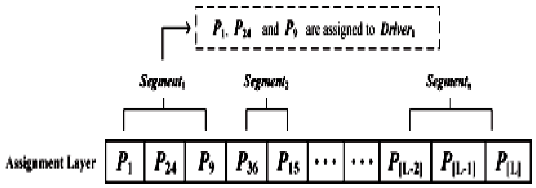


Fig3. Assignment layer for the proposed chromosome representation

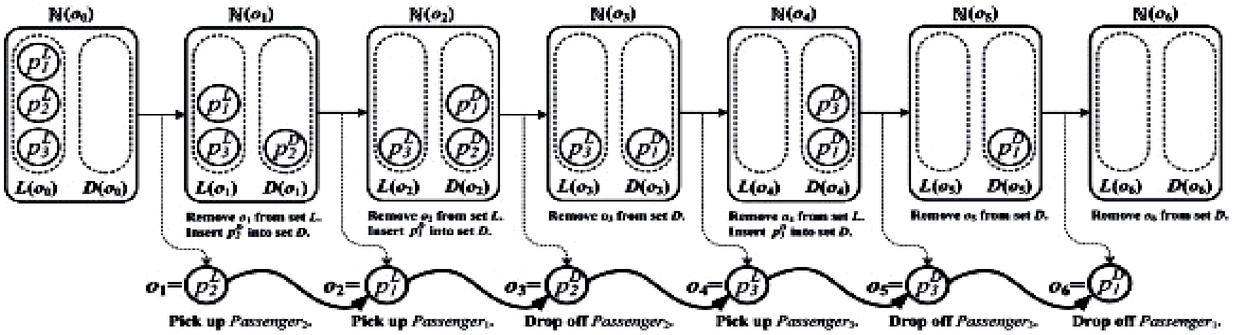
Fig. 3 illustrates an example in which Passenger1, Passenger24, and Passenger9 are assigned to Driver1, and Passenger36 and Passenger15 are assigned to Driver2. In the search space of the CSP, the number of segments can be dynamically modified to suit the quantity of drivers. Once the assignment layer generates each segment, the order in which drivers should pick up and drop off passengers is expressed by the implicit routing layer. For instance, carpool requests from drivers and passengers are received by the CGCS module of the AICS system, here upon one possible solution is encoded via chromosome representation. Fig. 4 presents an example in which Segment1 features three genes: a carpool consisting of Driver1, Passenger1, Passenger24, and Passenger9. The implicit routing of Segment1 dictates that Driver1 will pick up passengers according to a specific order (Passenger1, Passenger24, and Passenger9) and drop off each passenger according a different order (Passenger24, Passenger1, and Passenger9).

2) Greedy Population Initialization: To effectively is tribute the initial population in the solution space, the chromosomes that are first generation are arranged by designating a driver through the greedy strategy during the assignment process. The population of each generation can be described as Population g = {C1, C2, . . . , Cps}, where ps is the population size, and g represents the number of generations. The initialization procedure applies the proposed assignment strategy, which is a node-pair operator called distance based greedy heuristics. It initializes the segment of each driver according to the magnitude of the estimation values (ev) stored in the set EVi = {evi(j)}. The estimation values are expected to be minimal and are calculated by the estimation expression as

$$\mathrm{ev}^i(j) = c\left(d_i^L, p_j^L\right) + c\left(p_j^L, p_j^D\right) + c\left(p_j^D, d_i^D\right) - c\left(d_i^L, d_i^D\right)$$

Where c(w, x) is a travel distance cost value between two nodes, and dLi, dDi and pLj, pDj are the origin–destination pairs of drivers and passengers, respectively. An optimal value of ev close to zero indicates that the detour distance of the driver's route is slightly increased when the origin–destination pair is added to redirect the initial route, as shown in Fig. 5.

B. GE Module



1) Chromosome Evaluation: In order to evaluate the quality of the chromosomes in the population, the fitness function is used to determine the travel cost for each driver. The fitness value, Fk, of a chromosome is the sum of the fitness values of all segments as follows:

$$F_k = \sum_{i=1}^{m} f_i,$$

where Fk is the fitness value of the kth chromosome, fi is the fitness value of each segment in the kth chromosome, and m is the number of segments in the kth chromosome. To calculate the fitness value of a segment, we need to find the most efficient route for picking-up and dropping off passengers for each corresponding driver. The routing problem in a segment can be viewed as a graph problem:

G = (L,A) , (6) where L = 1, 2, . . . , 2Si is the index node set which represents the locations of passengers in the segment, and A = (1, 2), . . . , (w, x) is the set of arcs in the segment. The constraint is is that w _= x, ∀w ∈ L, ∀x ∈ L. The Cwx is

defined as the cost of traveling arc (w, x) between node w and node x. The initial location and destination nodes of the driver are defined by DL and DD, respectively.

According to the above definitions, the fitness function of each segment can be modeled as follows:

$$f_i = 1 \ / \sum_{j=1}^{(2S_i-1)} \left(C_{(j,j+1)}\right) + C_{(D_L,1)} + C_{(2S_i,D_D)} \ ,$$

where fi is the fitness of each segment, and C(w, x) is the travel cost of the arc(w, x).

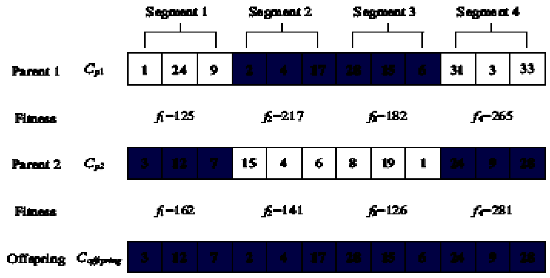


Figure 6. The elite-segment crossover procedure. The evaluation procedure will find the shortest route of each segment and subsequently record the routing result in the implicit routing layer of the chromosome. When the fitness value of a chromosome is larger, the routing result is better represented. Additionally, the fitness value can be used as a guide for the following chromosome selection procedure.

2) Chromosome Selection: After evaluating the initial population, the chromosome selection procedure begins To produce the new generation. In order for high-quality chromosomes to enter into the next generation and enhance the crossover procedure, we propose a two-phase selection method. The first phase involves sorting the chromosomes into a descending order according to their fitness values, and selecting those with the highest values in the population by using an elitist strategy. This strategy allows for the retention of chromosomes with the highest fitness values from one generation to the next, as well as ridding the population of low-level chromosomes. In the next phase, the probability value (Pp), of each top tier chromosome (Cp) is determined according to the ratio of the fitness value of each top-tier chromosome to the sum of the total fitness values of all top-tier chromosomes. This is shown as follows:

$$P_p = \frac{F_p}{\sum_{k=1}^{en} F_k} ,$$

where Fk is the fitness value of each top-tier chromosome, and en is the total number of top-tier chromosomes.

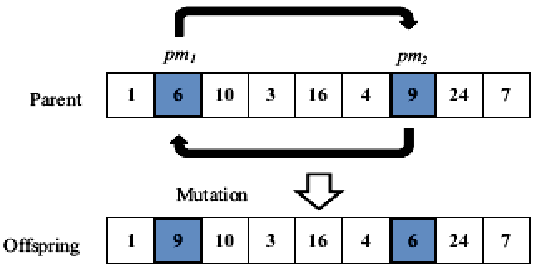


Figure 7. The chromosome mutation procedure

As described above, a chromosome with a larger fitness value has a larger probability of being selected for the following crossover procedure. For our proposed chromosome selection procedure, the determination of top-tier chromosomes is hastened using an elitism-based selection method followed by a roulette-wheel selection method.

3) Chromosome Crossover: After the optimal chromosomes have been selected, the chromosome crossover procedure is utilized to recombine the chromosomes of selectedparents to simulate the natural process of evolution and thus generate a more suitable carpool match by capturing and retaining the features of elite segments of the parent chromosomes. Two parent chromosomes (Cp1 and Cp2) are chosen for the crossover procedure from the population of the previous generation. The first parent chromosome (Cp1) is chosen from among the top-tier chromosomes through random selection. The second parent chromosome (Cp2) is selected from the remaining lower-tier chromosomes in the population, and is used to encourage mating diversity.

In order to find the best combination of chromosomes from the two chosen parents and produce a high quality offspring, an elite-segment crossover operation is proposed. The elite-segment crossover operation captures the features of the best segments of the parent chromosomes and passes these features onto a new offspring chromosome (Coff). The elite- segment crossover operation flow is described as follows: The two parent Cp1 and Cp2 are partitioned into m segments. Each segment has a fitness value (fi) which represents the travel cost to the corresponding driver. Let m denote the number of segments in the chromosomes, then for i = 1, . . . , m, the ith segment of Coff inherits the ith segment Cp1(i) of the top-tier parent chromosome and the ith segment Cp2(i) of the lower-tier parent chromosome. For example, the two parent Cp1 and Cp2 are partitioned into four respective segments, as shown in Fig. 6. Initially, the proposed elite-segment crossover procedure compares the fitness values of two Segment1 in Cp1 and Cp2, the procedure is repeated until the entireties of the segments are checked. To improve the quality of the offspring chromosomes, the segment with the better fitness value will be passed to resulting offspring. Segment1 and Segment4 of the offspring then are taken from Cp2, and Segment2 and Segment3 of the offspring are taken from Cp1.

4) Chromosome Mutation: In order to maintain population genetic diversity and thereby lessen the chances of getting stuck in a local optimum, the procedure of chromosome mutation is performed after the crossover procedure and is used to change the allocation of the passengers. Since chromosomes do not contain duplicate passengers, the use of a procedure called swap mutation is proposed. The implement of swap mutation is shown in Fig. 7. First, a passenger for the first mutation (pm1) is randomly selected. Then, a passenger for the second mutation (pm2) is and randomly selected. Finally, the positions of passengers pm1 and pm2 are exchanged in the chromosome, thus generating a new mutated offspring. After the procedure of chromosome crossover and chromosome mutation, the chromosome may be invalid when the passenger is assigned to more than one driver. Accordingly, it is necessary to repair the chromosome by replacing the invalid gene.

IV. CONCLUSION

Reducing the ecological footprint due to transportation is one of the key goals to cut the emissions of carbon dioxide. Advanced information technologies can heavily support this task, by suggesting to the users “smarter” mobility solutions among the various means of *sustainable* transportation. In this paper, this platform has been described. It is meant to overcome the limitations of current sustainable mobility solutions, by coherently integrating a wide range of new technologies. Among them, the Microsoft Azure platform allowed us to develop a Cloud-based back-end, where the computation of route arrangements can be done exploiting the massive elaboration power of the Cloud infrastructure.

The definition of CSP is also addressed for the application of practical system. Such a problem is successfully solved by our proposed AGCRMA, which is based on genetic algorithm and is composed of two major modules: an EI and a GE. The EI module can express the solutions to the CSP via chromosomes and utilizes proposed distance-based greedy heuristics to effectively generate initial population in the solution search space. The GE module is then able to find the optimum carpool route and matching results both accurately and promptly in accordance with the optimization of all objectives. Among that, the dynamic programming method is applied to promptly solve the origin–destination pair route problem within the evaluation process. The early stop option is additionally activated to facilitate the improvement of processing time.

V. REFERENCES

1. F. Y. Wang, S. Tang, Y. Sui, and X. Wang, "Toward intelligent transportation systems for the 2008 Olympics," IEEE Intelligent Systems, vol. 18, pp. 8-11, Nov. 2003.
2. Abstract | Full Text: PDF (896KB)
3. R. Barrero, J. V. Mierlo, and X. Tackoen, "Energy savings in public transport," IEEE Vehicular Technology Magazine, vol. 3, pp. 26-36, Sep. 2008.
4. Abstract | Full Text: PDF (2729KB) | Full Text: HTML
5. S. Tsugawa and S. Kato, "Energy ITS: another application of vehicular communications," IEEE Communications Magazine, vol. 48, pp. 120-126, Nov. 2010.
6. Abstract | Full Text: PDF (1520KB) | Full Text: HTML
7. S. Hartwig and M. Buchmann, "Empty Seats Traveling," Nokia Research Center Bochum, Feb. 2007.
8. R. Fagin and J. H. Williams, "A Fair Carpool Scheduling Algorithm," IBM Journal of Research and Development, vol. 27, pp. 133-139, Mar. 1983
9. Abstract | Full Text: PDF (606KB)
10. R. K. Megalingam, R. N. Nair, V. Radhakrishnan, and A. V. Vidyapeetham, "Automated Wireless Carpooling System for an Eco-Friendly Travel," Electronics Computer Technology, vol. 4, pp. 325-329, Apr. 8-10 2011
11. H. D. Nguyen, I. Yoshihara, K. Yamamori, and M. Yasunaga, "Implementation of an Effective Hybrid GA for Large-Scale Traveling Salesman Problems," IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 37, pp. 92-99, Feb. 2007.
12. Abstract | Full Text: PDF (432KB) | Full Text: HTML
13. H. C. W. Lau, T. M. Chan, W. T. Tsui, and W. K. Pang, "Application of Genetic Algorithms to Solve the Multidepot Vehicle Routing Problem," IEEE Transactions on Automation Science and Engineering, vol. 7, pp. 383-392, April. 2010.
14. Abstract | Full Text: PDF (926KB) | Full Text: HTML
15. *W. Ahn and R. S. Ramakrishna, "A genetic algorithm for shortest path routing problem and the sizing of populations," IEEE Transactions on Evolutionary Computation, vol. 6, pp. 566-579, Dec. 2002.*
16. Abstract | Full Text: PDF (1466KB)
17. for aircraft arrival sequencing and scheduling," *IEEE Trans. Intell.*
18. *Transp. Syst.*, vol. 9, no. 2, pp. 301–310, Jun. 2008.
19. [17] M. Mesbah, M. Sarvi, and G. Currie, "Optimization of transit priority in
20. the transportation network using a genetic algorithm," *IEEE Trans. Intell.*