A Multi-platform Game Theory Based Taxicab Assignment System for Independent Drivers

Ruibin Bai, César D. Velandia-Briñez, Jason A.D. Atkin, Graham Kendall, John Woodward

***Abstract*—Taxicab hailing is the most commonly used method for hiring vehicles across the world. In some cities, taxi drivers are hired solely by customers flagging them on the road and typically booking systems are ineffective or not readily available. New technologies, mainly novel communication systems, are being applied to deal with situations like this. To date, however, communication between vehicles is limited or doesn't exploit the potential of these approaches; it is used only to book vehicles as requested by passengers. It is often the case that drivers compete for customers or miss opportunities due to lacking details of the demand information. Access to mobile devices proposes an interesting method for information dissemination including all involved parties. In our proposed system, mobile devices are used to assist the matching of taxi drivers and hailing customers based on the principles of game theory, which produces a stable assignment structure with benefits for the whole system, as well as for individual drivers. In this work, we discuss the design and implementation of such system, complementing the model described in our early work. The system represents the first decision support system designed for dispatching coordination between independent taxicabs.**

***Index Terms*—Taxicab assignment, Game theory, Stable marriage problem, Street hailing, Mobile devices**

I. Introduction

MULTIPLE intelligent systems have been devised during the last decades to cope with increased demand for transportation in cities. In particular, the use of hand-held devices, e.g. smart-phones, has revealed a wide range of possibilities for communication between parties and organisation of scarce resources. Most of the effort has been concentrated on mass transportation modes such as buses, subways and trains. However, hired vehicles or taxicabs represent an industry with large impact on traffic operations. The assignment of taxicabs is primarily based on two modalities; operators allocating company affiliated drivers to calling passengers or street hailing.

While in some cities, taxicabs are required to be part of taxi companies: in other places it is common to find unaffiliated cars or independent drivers. They are hired mostly by means of street hailing or taxi stands. While the latter is an effective way to deal with empty taxi journeys and additional fuel consumption, it demands sometimes considerable time queuing at the stand. Therefore, independent drivers usually prefer to look for customers in areas of the city with a history of high demand, which can contribute to congestion and is not always a cost-effective method. In recent years, customers are using mobile technologies to assist them in the process of taxi hailing [2] [3]. Existing applications act as interfaces to mediate between company operators, drivers and customers. It is clear that previous approaches work under the assumption taxicabs are affiliated to taxicab companies. The problem of assigning unaffiliated taxi drivers to customers has not been addressed before.

This paper is concerned with the allocation of independent taxicab drivers to hailing customers, which was previously presented in detail in [1]. A multi-platform taxi assignment system has been developed to provide decision support for independent taxi drivers, based on a game theoretic model. To deliver information to drivers and users, a mobile application was implemented allowing two-way communication and to find an optimised match between vacant taxicabs and hailing customers. In this paper, we discuss the implementation of such a system and detail its composition and some practical issues. The matching algorithm was tested using a simulator developed as part of this project.

II. The taxicab assignment problem

The problem of assigning scarce assets is found frequently in operational research. Allocating vacant taxis and hailing customers is done basically considering the distance between each other and the time of the request [4]. Taxi companies provide a centralised operator system that broadcast the customer's location; drivers, based on their knowledge of the local area, decide whether to attend the request or not. This has been widely discussed in the dial-a-ride problem [5] [6]. There are two main issues with this approach in this context. First, the vehicle attending to the customer might not be the closest, since the taxi driver knows his distance to the customer but not the distance or avaliabiltiy of other drivers. Second, there is no consideration of other customers' requests; every call is effectively treated independently, in which case the the nearest available vehicle is dispatched. The role of the operator/dispatcher is fundamental in this model. Latest dispatching systems are provided with more sophisticated tools, such as global positioning sytems (GPS) to

R. Bai is with the Division of Computer Science, The University of Nottingham Ningbo China, 315100 e-mail: ruibin.bai@nottingham.edu.cn

C. D. Velandia-Briñez is with the Division of Computer Science, The University of Nottingham Ningbo China, 315100. e-mail: cesar.velandia@nottingham.edu.cn

J.A.D. Atkin is with the School of Computer Science, The University of Nottingham, Nottingham, NG8 1BB, UK.

G. Kendall is with the School of Computer Science, The University of Nottingham, Nottingham, NG8 1BB, UK.

J. Woodward is with the Division of Computer Science, The University of Nottingham Ningbo China, 315100.

This research is supported by "Qian Jiang" Talent Initiative Program (QJD0802001) and Ningbo Sci.&Tech. Bureau (2008B10040).

determine positions for each vehicle affiliated to a company. While this system works well in some situations, it still relies on the dispatcher's judgement and assumptions made by drivers. Another aspect that must be considered is the role of the driver in the market. In some countries, taxi are not required to affiliate to a company or simply do not receive requests from a centralised system. Nevertheless, it is desirable to provide a way to integrate independent taxi drivers and customers to avoid increased expenses to both parties and reduce urban traffic.

Taxicabs can be assigned to customers using a game theoretic model. Taxicabs are resources to be assigned to customers. The assignment process is determined by the distance between each other; however, greedily assigning the closest customer is not always the best overall choice. To approach this problem the proposed system was modelled in similarity to the stable marriage problem discussed in [7]. In this model, women are paired to men based on their personal preferences. The algorithm proposed finds a stable matching, i.e., an assignment in which no pair will benefit further from choosing a different partner. In a similar manner, taxicabs are assigned to hailing customers, avoiding competition and producing a matching structure that will benefit all the participants in the long term.

Even though taxicabs can be booked on-line or by telephone, hailing is still the most common way to hire a taxicab as this method does not require any previous action from the customer. The visibility between customer and driver and the number of taxis are important factors in this process. Clearly a customer cannot hail a taxi if they cannot see one another In recent years it has been proposed that taxicab operations can be enhanced using hand-held GPS devices [8][9]. Particularly, smart-phones provide multiple advantages as they have the capacity to pinpoint a user's position accurately and communicate it with other systems. A system utilizing such devices to hail a taxicab is therefore viable and provides the following benefits:

- provides accurate locations of users and available taxicabs
- requires less attention of the driver or customer
- enables more participants to be considered during the matching process, obtaining more practical allocation structures
- discourage drivers from racing and competition for customers, if the matching system is fair
- increases awareness of waiting customers and available taxicabs, therebty reducing waiting times and vacant taxi mileage
- accumulates information which can be used by road transport planning authorities and researchers

These advantages provide an interesting prospect and reveal the importance of a solution in this area. In order to provide a fair and stable assignment of taxicabs to passengers, we have adopted a game theoretic model and solved the problem by using a well-known algorithm, similar to a classic resource assignment problem.

III. Stable Marriage and Taxicab Assignment

The stable marriage problem was described by Gale and Shapley [7], as the problem of assigning resources based on preference structures. Women and men rank each other based on their personal preferences, that is, each participant has a most and a least preferred counterpart and vice versa. The algorithm proposed in [7] achieved a stable matching, which can be defined as the assignment that benefits the whole system and in which no participant will benefit from disregarding the matchmaker's recommendation. The Gale-Shapley algorithm starts with men proposing to each women in their preference structure (suitors or proposers), while women decide whether to keep their current match or discard it to accept the new match (acceptors). After every unpaired suitor has proposed to all the women in his preference structure, the resulting matching is stable. This algorithm has been extended to other scenarios, matching resources in hospitals (patients, nurses, rooms, doctors) and universities (students, time slots, rooms, equipment) among others [7]. Our system applies this same principle to the problem of assigning independent taxicab drivers to customers.

The motivation of this design is to address the customer driver dilemma. If every driver tries to find customers as soon as possible, in many instances picking up the closest customer is not the best choice. This happens due to other drivers competing for the same customer. A matching that considers other taxis and customers will amost certainly provide better assignments, reduce the amount of travel by empty taxis and avoid competition between drivers. Fig. 1 displays a simple scenario in which the same customer is the closest option for two drivers. A better strategy is to assign customers based on a criterion that takes into account of other vehicles and drivers and produces a stable matching.

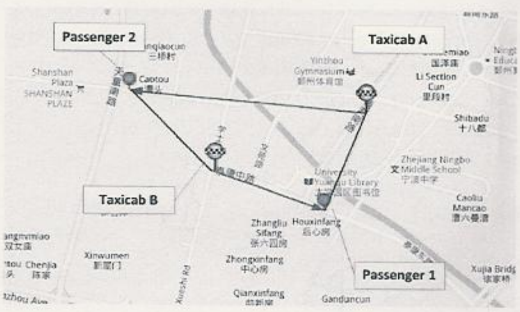


Fig. 1. The driver passenger dilemma. Passenger 1 is closer to Taxicab A, however, picking up Passenger 2 is a better strategy since Taxicab B is closer to Passenger 1 and will likely arrive first.

IV. Design Principles

To provide better taxicab services which takes into account non-affiliated drivers, we have envisioned a system in which users (i.e., independent drivers, taxi customers) interact with each other using mobile devices with GPS. Other potential information consumers are control agencies and research centres which will have access to information in real-time through the internet. Therefore, this system was designed to allow multiplicity of users and interfaces.

In order to collect the information and provide a convenient tool to hail taxicabs, the system contains a dedicated hailing application, available for both customers and drivers. The application was developed for smart-phones with a connection to 3G networks and location information via the mobile network or GPS. The development of the application included an unobtrusive notification service and a map-based interface, with intuitive access to relevant information based on the type of users.

Using a system that provides limited, but opportune information was of great significance during the development of this system. Taxi drivers need to pay full attention to the task of driving, thus a suitable application must require minimal distraction bydisplaying meaningful information to its users.

Another important consideration that we have made is the transparency of the system in terms of the assignment procedure. Users experience negligible delays even if the number of vehicles and customers to assign is large. We have accomplished this by dividing the operation of the network under study in manageable zones, providing a maximum travelling distance to pick up a customer and limiting the number of matching users (e.g. drivers and customers).

Maintaining information about the location of drivers and customers is required in this system. However, this information must be stored and distributed safely. For this purpose, we designed applications that display only information of the actual user (driver or customer) and the proposed match. This simplification, apart from providing a clear interface, also maintains customers' privacy and encourages compliance with the system, ensuring that only one taxicab will attempt to pick up one customer after the application has been initiated.

Our assignment system runs on a server connected to the internet, obtaining updates and distributing information using standard web services. The system was designed to be scalable and all its components can be extended or reused. Exposing the assignment system functionality using web services provides the opportunity of integration with other systems or extensions to include other information producers and consumers.

V. Proposed Architecture

To tackle the problem of ineffective assignment of independent taxicab drivers to hailing customers, we have developed an architecture in three tiers to facilitate communication between both parties (see figure 2). Our model complies with the structural design principle aforementioned. The front-end of the application contains two user interfaces. A web interface, developed to present the state of the network, was developed using Google's mapping service and implemented with a scripting language (and the jQuery library) to retrieve data from the assignment server. The development of this interface, besides providing a clear representation of the matching system, also has the purpose of displaying vehicles and customers moving around the map. This simulation is based on a stochastic formula, including a probability of maintaining the travel direction. The second interface in the front-end includes a smart-phone application for Android OS. Two versions (i.e., for both drivers and customers) were developed. After a simple registration process, users of any type (drivers and haling customers) can update their status and begin broadcasting their location to the server. The smart-phone application has two main purposes; to update the user's location to the server and to report the location of the matched customer or driver.

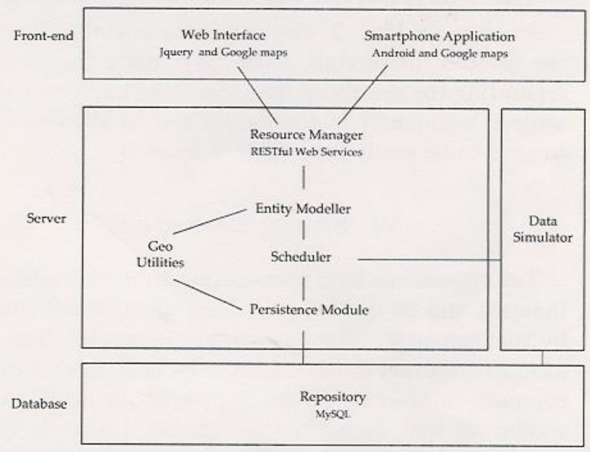


Fig. 2. Multi-tier architecture for the proposed taxicab assignment system

In order to connect multiple devices in an effective way, and allow for further extensions of this system, web services were used to interact between server and clients. Updates in both directions require certain frequency, and the data models are made clear and simple. These services were implemented using the RESTful (Representational state transfer) model which allows for transmission using XML format and a simple but effective set of operations. On the server side, an entity modeller is in charge of creating data structures for the assignment module. A module containing geographic utilities helps with tasks such as distance computation, and coordinate normalisation. The data structures generated by this model is then passed to the game-theoretic assignment system, which requires location details of the customers and the drivers, distances between each other (which will be used to compute the initial preference structures). The distance structure is biased (favourably or unfavourably) using scores maintained in the data repository for recurrent users. The structure is obtained using a simplified Gale-Shapley algorithm and requires auxiliary structures to keep the current best match and the number of times a customer has been rejected (during the algorithm execution). The resulting solution contains a stable assignment between users, where no driver or customer will benefit from disregarding the system's recommendation.

The persistence module connects to the data repository to retrieve information from the historical compliance record and to store position updates and assignments made by the assignment system. A recommendation made by the assignment module and accepted by both parties is called *contract* in our system. Contracts contain information about the two contractors (i.e., driver and customer), initiation and completion times. The repository stores information of each customer and driver including their current score. Our design maintains a record of each individual location updates (movements of taxis and customers), with the appropriate identifier, coordinates, update time, travelling direction and current status (e.g., hailing).

As show in Fig. 2, the simulator connects directly to the assignment module. We have made the decision of separating the simulator from the main implementation to achieve modularity in the design and to allow other data sources to be easily connected if desired.

VI. System Capabilities

The system has been tested using an initial dataset of 20 taxicabs and 30 customers. Their updates are controlled by the simulator. The dataset was generated from a uniform distribution in the central area of Ningbo, China. To compare the results of the approach, an initial run was conducted with an ad-hoc assignment (taxicabs were assigned as the requests were made with no considerations of distance, other vehicles or hailing customers). This can be considered as a rough approximation of the current street-hailing approach in real world. The results obtained were compared with another approach in which contracts were obtained using the game-theoretic assignment module. The results from these runs are shown in Fig. 3. To compare these results we also obtained the optimal matching for the same problem as follows.

Optimal matching

When taxi drivers work in a collaborative manner (i.e. divers are not seeking maximum personal gain but instead cooperate between each other in order to achieve a socially maximum gain), it is possible to achieve an optimal schedule so that the total vacant taxi mileage required to collect all of the customers is minimised. Recently Seow et al. [11] proposed a multi-agent system to coordinate the collaboration between taxis. However, due to the natural of the multi-agent system, the solution is often sub-optimal. In fact, this problem can be formulated as the following linear assignment problem.

Denote C be the set of customers and T be the set of taxis. Let d_{ij} be the distance between customer $i \in C$ and taxi $j \in T$ and x_{ij} be the assignment decision variables (i.e. $x_{ij} = 1$ if customer i is assigned to taxi j and 0 otherwise).

TABLE I

Travel distances for the test dataset with ad-hoc, co-ordinated and optimal assignments

Assignment	Travel distance (km) Average	Total
ad-hoc	5.137	102.749
stable	1.602	32.058
optimal	1.466	29.332

$$\min \quad \sum_{i\in C}\sum_{j\in T} d_{ij}x_{ij} \tag{1}$$

subject to

$$\sum_{j\in T} x_{ij} = 1 \quad \forall i \in C, \tag{2}$$

$$\sum_{i\in C} x_{ij} = 1 \quad \forall j \in T, \tag{3}$$

$$x_{ij} = \{0,1\}. \tag{4}$$

This problem can either be solved using the well-known Hungarian Algorithm [10] or by a linear programing solver since the constraint matrix is *totally unimodular*. The optimal matching is unsuitable for our system since independent drivers may not want to sacrifice their individual interest for an overall-best group interest. However, for the purposes of comparison and analysis, it is included in this paper. Table I presents the results obtained by the stable matching algorithm in comparison with the optimal assignment. The total distance in the stable matching is slightly higher than the distance resulted from the optimal matching, which may be interpreted as the *price of anarchy* in game theory [12].

As summarised in table I considerable improvements were obtained after using the game-theory assignment, even without restrictions on the maximum travel distance between customers and vehicles. In case such restriction is applied, the system will keep unpaired customers in a waiting list, until the next execution when new unpaired vehicles will be available.

VII. Implementation Details

This section contains the most relevant details of the assignment system. The items described here include decisions made in the development of the server modules and the mobile applications.

A. Preference structures

The Gale-Shapley algorithm solves the stable marriage problem using two preference structures to match women with men. The algorithm can be implemented having one or the other taking the matching decision, for example, the classic version supposes that each round men propose to women based on their preference structure, then each round women select the suitor for which they have higher preference based on their own structure. Conversely, men could be proposed to by women as well, but resulting in a same solution.

To match taxis and customers, the basic preference structure is based on the distances between them. Therefore, initial preference structures are the same for both taxis and customers. This unique preference structure contains the distance between parties at the moment of the assignment. This simplification reduces one step in the assignment algorithm in which assigned customers are always the best choice and no further check is required. In this problem, passengers "propose" to taxis. Then taxis decide whether to pick up the passenger or not. Note that these are not decisions by users, but part of the abstraction used in the algorithm itself.

An extension of these structures includes a penalty/reward system. The preference structure can be modified based on rewarding drivers with a good service record or penalizing drivers not complying with the system, such as breaking accepted contracts or pick up customers not assigned to them. . After implementing this extension, the preference structures for taxis and passengers are biased based on these historical records. After the completion of the contract, if successful, recurrent users receive awards, otherwise, they receive penalties based on who broke the contract.

B. *The hailing contract*

Each user is provided with information relevant to his contract. In other words, taxicab drivers do not have access to other vacant taxicabs positions or even all the hailing customers in their areas. This will avoid drivers competing to pick up passengers. Customers will have access to the location and position of the vehicle assigned only as the contract is proposed. Both driver and passenger must agree with the assignment, which will remain active until the customer is picked up by the taxi. This design provides a simple sequence for the involved parties, i.e., status update and contract confirmation. Customers and drivers in a contract together can track the position of each other using the application's interface. This therefore implies that information transmitted is lower.

Contracts are defined by the procedure described in Fig. 4. The system firstly obtains the matching participants, collecting all the drivers on duty and customers on demand. This is determined by the initiation of the mobile application and the status updates, which trigger the initiation of the location updates. When the assignment system determines a stable matching, it notifies each of the paired participants with proposed contracts. The information of the paired participants is displayed briefly to assess the decision of the user. If the contract is accepted, then it is initiated. Otherwise, the users who did not accept the contract are included in a waiting list to be considered in the next assignment run. Initiated contracts remain active, updating positions of the paired users to the server which distributes them to the corresponding match. When the driver arrives at the customer's location, the contract is finalised automatically. This simple process allows to maintain and distribute the positions of the interested parties, until the hailing process is completed.

C. *Assignment Periodicity*

Two mechanisms can be used to determine how often the assignment system will match drivers and customers. The first mechanism is time-based. A pre-set time limit (e.g. 3 minutes) ensures that the assignment process is triggered with enough frequency to provide an acceptable service rate. The other mechanism to assign users could be the number of vehicles and customers requiring allocation. Once the user pool reaches a number of customers hailing and vacant taxicabs, an execution is carried out to increase the responsiveness of the system. We used the first method in our simulation.

D. *Update Frequency*

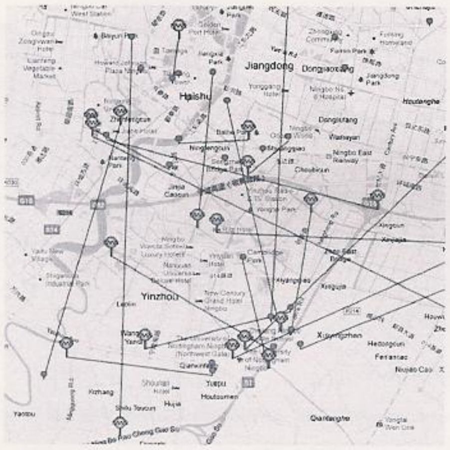
Drivers and customers positions are updated while they report their status as *available* for taxicabs and *hailing* for passengers. After a contract has been initiated, updates are still passed to the assignment system, which retransmits these coordinates to the other parties. As soon as the contract terminates (i.e., the driver picks up the passenger), the mobile clients stops broadcasting. The frequency of these updates is increased during the duration of the contract and since vehicles are capable of moving faster than customers, their position must be updated considerably more frequently than customers. We have chosen an initial update period of 5 seconds for vehicles and 10 seconds for customers.

E. *Distance considerations*

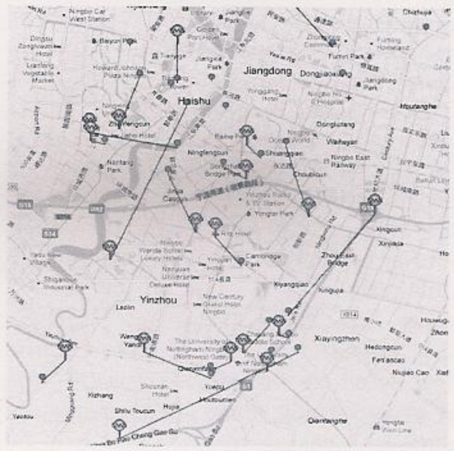
The physical distance between a hailing customer and a vacant vehicle can be obtained directly using Euclidean geometry. These simplifications are practical choices but do not represent the actual distance or time required by a vehicle to reach a potential passenger. For instance, if a customer hails a vehicle on the other side of the road, the real distance between taxicab and customer is determined by the distance to the next permitted U-turn. This scenario is one of many situations in which distance does not represent the actual travel trajectory or time to reach the customer. However, it represents a practical approximation to the scenario and we have decided to assume direct distance as the decisive preference factor.

F. *Zone assignments*

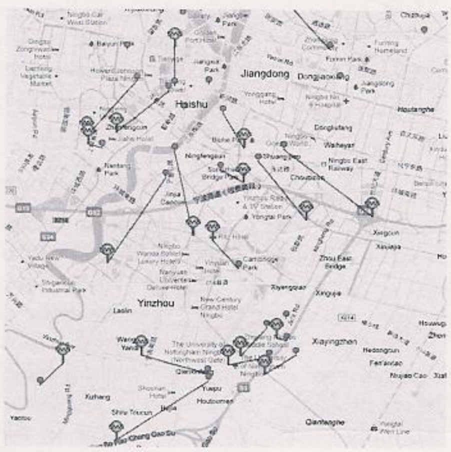
The system divides a city map into zones which permits better control (by better "control" do you mean it provides more practical solutions???) over the assignment process during the matching process. Vehicles are assigned to customers within the same zone or adjacent zones. The reason for this is that customers and vehicles near the boundaries of contiguous zones might be closer to each other than users in their respective zones. This feature allows us to discard feasible but impractical matching with distant customers. The city's breakdown into zones allows for better



(a) ad-hoc



(b) co-ordinated,stable assignment



(c) optimal assignment

Fig. 3. Taxicabs and customers assigned in a section of Ningbo City. In Fig. 3a vehicles are assigned based on the order of arrival of each vehicle, in Fig. 3b assignment is based on a game-theoretic algorithm. Fig. 3c is the optimal assigment based on the Hungarian algorithm.

control but must not constrain the vehicle assignment if the distance to a hailing customer is the shortest. Another practical distance implementation consideration includes setting a maximum travel distance limit, which will discard other impractical assignments (within and between zones).

In order to improve the response of the system, the system maintains user information organised by zones and keeps a copy of the most recent location and status of each one in memory. If a driver and a customer have initiated a contract, this information is maintained in memory as well.

This zone proximity allows us to compute matches more rapidly. These records are maintained and updated in the assignment system repository by a background process.

G. Identifiers

To identify customers, the mobile number from which the service is request is used. To identify taxicab drivers, the system requires a registration plate of the vehicle as well as the taxi driver's mobile number.. This allows us to keep track of drivers and vehicles, in the scenario where drivers share taxicabs on different shifts.

[I'm not quite sure why "memory" is important]

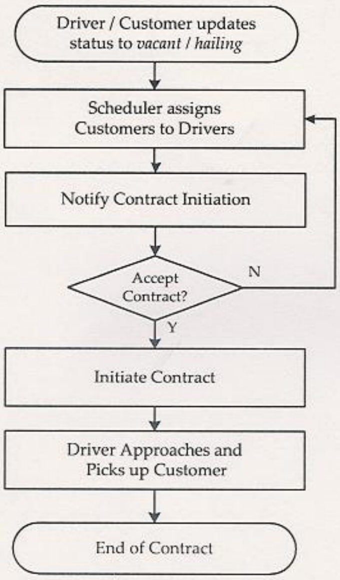


Fig. 4. Life-cycle of the hailing contract

H. Additional implementation details

The mobile application used in this system was implemented using Android phones and Google maps for Android OS, which allow for tests using a wide range of different devices. The operation system version was Android OS 2.2 and Google maps for Android API version 3.2. The server side was implemented in an Apache Tomcat server 7.0.12, running on a machine with 4 GB of RAM and a 2.4 GHz processor. Java was used for the server side implementation and the smart-phone application.

VIII. Future work

The proposed system can be extended in terms of the server modules and the mobile application. Statistics related to the use of this system (e.g., average waiting times, number of hailing contracts completed) could be extracted and presented to users to help them understand better their own practices and the effectiveness of the system. Also, we would like to extend the application to other operative systems.

From the server side, further improvements could be achieved by including a more efficient way to maintain the zone-based structures in memory, and scale the system to larger geographical areas. We have been considering a parallel version of the assignment algorithm capable of matching large numbers of participants without delays.

IX. Conclusion

We have outlined and implemented a system for taxicab assignment for independent drivers. As far as we know, this is the first such a system to provide automated coordination for independent drivers. The problem is modelled and solved based on the well-known Gale-Shapley algorithm for the stable marriage problem. The proposed system includes a wide range of user interfaces and is capable of managing other connections. The systems was initially tested using simulations.

The assignments obtained by our system provides noticeable improvements in terms of travel times (near to the optimal solution), which in turn, is reflected in reduced waiting times and resource consumption. Devices with access to wireless network and GPS can be used to collect positions of the participants and assist the hailing process. This system is cost effective and provides a simple solution to a process that has received very little attention and can provide great improvements in urban transportation and everyday activities. The system complements well current centralised taxi dispatching systems designed for companies with large number of (dependent) taxis.

References

[1] R. Bai, J. Li, J. A. D. Atkin, and G. Kendall, "A Game Theoretic Approach for Taxi Scheduling Problem with Street Hailing," in *Proceedings of the 5th Multidisciplinary International Conference on Scheduling: Theory and Applications*, J. Fowler, G. Kendall, and B. McCollum, Eds. Arizona, 2011, pp. 161–172.

[2] Z. Liao, "Real-time taxi dispatching using Global Positioning Systems," *Commun. ACM*, vol. 46, no. 5, pp. 81–83, May 2003.

[3] J. M. Salanova, M. Estrada, G. Aifadopoulou, and E. Mitsakis, "A review of the modeling of taxi services," *Procedia - Social and Behavioral Sciences*, vol. 20, no. 0, pp. 150–161, 2011.

[4] M. von Massow and M. S. Canbolat, "Fareplay: An examination of taxicab drivers' response to dispatch policy," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2451–2458, Mar. 2010.

[5] E. Feuerstein and L. Stougie, "On-line single-server dial-a-ride problems," *Theoretical Computer Science*, vol. 268, no. 1, pp. 91–105, Oct. 2001.

[6] J.-F. Cordeau and G. Laporte, "The dial-a-ride problem: models and algorithms," *Annals OR*, vol. 153, no. 1, pp. 29–46, 2007.

[7] D. Gale and L. S. Shapley, "College Admissions and the Stability of Marriage," *The American Mathematical Monthly*, vol. 69, no. 1, pp. 9–15, Jan. 1962.

[8] J. Yu, X. Zhou, and H. Zhao, "Design and Implementation of Taxi Calling and Dispatching System based on GPS Mobile Phone A Research for LBS Application Teaching Case," in *ICCSE 2009: Proceedings of 2009 4th International Conference on Computer Science & Education*, New York, NY, USA, 2009, Proceedings Paper, pp. 1163–1169.

[9] J. Villar, A. Otero, J. Otero, and L. Sánchez, "Taximeter verification with GPS and soft computing techniques," *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, vol. 14, no. 4, pp. 405–418, Feb. 2010.

[10] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network flows: theory, algorithms, and applications*. Prentice Hall, 1993.

[11] K. T. Seow, N. H. Dang and D.H. Lee, "A collaborative multiagent taxi-dispatch system", *IEEE Transactions on Automation Science and Engineering*, vol. 7, no. 3, pp. 607–616, 2010.

[12] J. Corbo and D. Parkes. "The Price of Selfish Behavior in Bilateral Network Formation", in *ACM Symposium on Principles of Distributed Computing*, pp. 99–107, 2005.