Title	Taxi dispatching and stable marriage
Author(s)	Kümmel, Michal; Busch, Fritz; Wang, David Zhi Wei
Citation	Kümmel, M., Busch, F., & Wang, D. Z. W. (2016). Taxi dispatching and stable marriage. Procedia Computer Science, 83, 163-170. doi:10.1016/j.procs.2016.04.112
Date	2016
URL	http://hdl.handle.net/10220/47119
Rights	© 2016 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 83 (2016) 163 – 170

Procedia Computer Science

The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016)

Taxi dispatching and stable marriage

Michal Kümmel[a,*], Fritz Busch[b], David Z.W.Wang[c]

[a]*TUM CREATE, 1 CREATE Way, #10-02 CREATE Tower, Singapore 138602*
[b]*Chair of Traffic Engineering and Control, Technical University Munich, Arcisstraße 21, München 80333, Germany*
[c]*School of Civil and Environmental Engineering, Nanyang Technological University, N1-01c-74, 50 Nanyang Avenue, Singapore 639798*

Abstract

This study explores the challenge of the dynamic dispatching of taxis to the immediate passenger booking requests. In particular, the study leverages on a stable marriage assignment algorithm and applies it for dispatching taxis to passengers. The stable marriage algorithm was developed initially for matching men and women according to their preferences in polynomial time. The results of the custom built simulation model show that the taxi dispatching strategy based on the stable marriage matching improves the taxi operation performance in all observed indicators (taxi profit, number of served passengers, not-occupied and total taxi mileage and passenger waiting time) as compared to the standard first-come, first-served strategy.

Peer-review under responsibility of the Conference Program Chairs

Keywords: Real-time taxi dispatching; Stable marriage assignment algorithm; Taxi; Vehicle routing; Simulation model

1. Introduction

This study focuses on the challenge of dispatching taxis to immediate passengers booking requests. Traditionally, new booking requests are assigned to taxis sequentially on a first-come, first-served (FCFS) principle, with the nearest available taxi dispatched in response to each new passenger booking request. This vis-à-vis assignment is straightforward and easy to implement but may produce unsatisfactory assignment of taxis to passengers, as illustrated in Fig. 1 as follows.

* Corresponding author. Tel.: +6566014026.
E-mail address: michal.kuemmel@tum-create.edu.sg

Peer-review under responsibility of the Conference Program Chairs
doi:10.1016/j.procs.2016.04.112

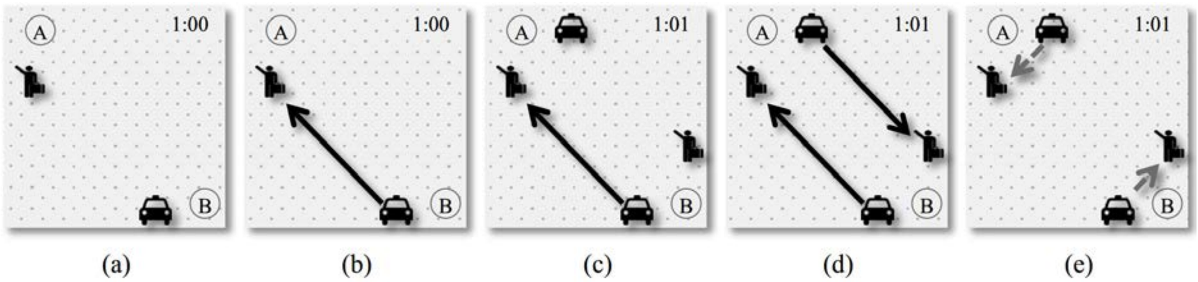

Fig. 1. (a-e) Illustration of sequential dispatching strategy drawback and improvement potential by simultaneous assignment of requests.

Supposing that a passenger at position A requests a taxi (Fig. 1.a), the nearest available taxi is at position B and is assigned to pick up the passenger (Fig. 1.b). A second passenger requests a taxi to transport him from position B (Fig. 1.c). Now the nearest available taxi is at position A. Consequently, both taxis have long drives - in the opposite direction along the same route - to pick up their passengers (Fig. 1.d) and both passengers wait longer than necessary. They would benefit from exchanging their assignments (Fig. 1.e).

This study investigates the application of a promising matching strategy based on stable marriage algorithm for taxi dispatching and discusses trade-offs of this dispatching strategy for taxi fleet operations. The following considerations are used to formalize this problem.

2. Problem formalization and assumptions

The taxi dispatching problem can be considered as an extended form of a Vehicle Routing Problem (VRP). The classical VRP by Dantzig and Ramser asks "What is the optimal set of routes for a fleet of vehicles to traverse in order to deliver to a given set of customers?"[1] The classical problem assumes that customers, fleet size and vehicle capacities are known upfront. Further, it requires the vehicles to begin and end in a central depot, to fulfil all the customer demands, not to violate vehicle capacity constraints and predetermined maximal route length. The overall objective is to minimize the total cost of the routes.

The taxi dispatching problem asks: “Which taxi should be dispatched to which passenger booking request?” The taxi operator either accepts or rejects booking requests if there are not enough taxis available. Once confirmed, the booked trip must be served. This is how the dispatching problem differs from the classical vehicle routing problem.

Furthermore, it is assumed that passengers desire to be transported from origin to destination as soon as possible from the time they make a booking request. Therefore, the requests are immediate and the degree of dynamism equals one.[2] The requests cannot be postponed, unlike for example in the study by Angelelli et al.[3] Moreover, passengers are not willing to share a taxi with anyone else and are only willing to wait for taxis for a limited amount of time. Taxis do not originate and end in one depot, but are geographically distributed at locations where shifts begin. Thus, the dispatching problem could be called: MDMVCDVRPPDTWDCR - Multi Depot Multiple Vehicle Capacitated Dynamic Vehicle Routing Problem with Pickup and Delivery Time Windows and Deniable Customer Requests.

The taxi dispatching problem has been studied from two major viewpoints in the taxi research literature: Rule-based sequential taxi dispatching (first-come, first-served, one-by-one) and simultaneous taxi dispatching (concurrent assignment of taxis to passenger booking requests.

Most of the initial research focused on the rule-based dispatching rules. In one of the first studies, Bailey and Clark[4] investigated efficiency of basic taxi dispatch rules: (1) closest free taxi, (2) closest occupied taxi or (3) the taxi that is free the longest. Following research investigated details of these rule based first-come, first-served assignments such as: whether it is more suitable to assign the geographically nearest taxi Jianxin et al.[5], Chang & Wu[6] and Grau et al.[7] or the taxi which can reach the passenger the fastest Lee & Wu[8], whether to consider real time traffic[9] or whether to use fuzzy logic[10] or alter the rule based strategies if there are more requests than taxis or vice versa[11].

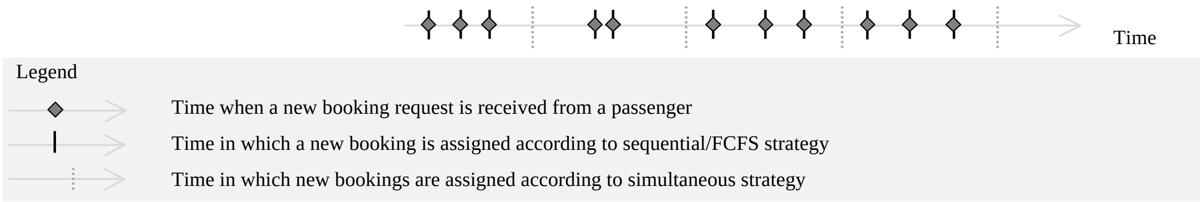

Fig. 2. Sequential and simultaneous assignment times of new booking requests.

Despite these partial improvements these studies were not able to overcome the fundamental limitation of sequential assignments of worsening the schedule quality over time. A solution is to buffer more booking requests and assign them simultaneously as illustrated in Fig. 2. Studies from some authors use the concept of simultaneous assignment of buffered requests[12,13,14] and they differ in the particular way the assignment is done.

Most studies focus only on the interests of the taxi operators and not the interests of passengers. One of the few studies that consider the interests of passengers, despite not focusing on taxi dispatching but street hailing segment, is the study by Bai et al.[15]. They noted that in many Chinese cities, dispatching systems are not used and taxi drivers have difficulties to find passengers. They suggested addressing this issue by an advisory system, which would be based on principles of stable matching. They compared this system with a virtual vis-a-vis dispatching system. The results, however, were not promising. Despite an average not-occupied taxi mileage improvement, the average total mileage and the average passenger waiting time increased. Nonetheless, their approach inspired this study to apply the stable marriage algorithm in the taxi dispatching context and answer some of the loose ends in order to estimate the potential of this algorithm for the taxi dispatching problem and learn how some aspects of the stable marriage algorithm implementation influence the efficiency of taxi dispatching. The following sections introduce the proposed dispatching strategy based on the stable marriage algorithm, describe the simulation model and the experimental set-up and present and discuss the results.

3. Methodology of the investigated dispatching strategy

The proposed dispatching strategy leverages on the fundamental principle of simultaneous assignment of many booking requests to taxis. This is done in regular time intervals in real time using the stable matching algorithm. If there is no taxi that can serve the booking request, the booking request is rejected and the passenger attempts to book again. If there are no new booking requests in a given time interval, no assignment is made.

An assignment using the stable marriage algorithm, which matches two parties using the Nash equilibrium, is suitable for taxi dispatching because it is a real time simultaneous assignment method based on bilateral preferences. The algorithm aims to achieve a system-wide stable assignment of two parties at each decision epoch. Some authors such as Moore and Passino[16] claim that a stable solution may be more useful than the close-to-optimal one in the presence of disturbances such as communication delays or if short computation time that reduce the quality of assignment by traditional heuristic approaches.

The solution algorithm to the stable marriage problem, proposed by Gale and Shapley[17], has been applied in various situations, such as the matching of students and schools, doctors and hospitals and kidneys and patients. The objective of the problem is to match both parties in a stable fashion according to their preferences. The outcome guarantees that nobody has an incentive to change the existing matching (in the sense of Nash equilibrium). The algorithm facilitates the transition of competitive individualistic preferences into a stable situation.

In the original Gale-Shapley algorithm, men propose to women. Men and women are assumed to have preference lists about the other sex, as illustrated in Fig. 3. Men propose to women according to their preferences and women response with either acceptance or rejection of men proposals according to their preferences. The matching process runs in rounds as long as there is an unengaged man who proposes to a woman.

How does the taxi dispatch problem resemble a marriage problem? Table 1 summarizes key differences and similarities of the general stable marriage problem and taxi dispatch problem, which are discussed in the following paragraphs.

Men	Preference list
A	a, b, c, d
B	b, a, c, d
C	a, d, c, b
D	d, c, a, b

Women	Preference list
a	A, B, C, D
b	D, C, B, A
c	A, B, C, D
d	C, D, A, B

Fig. 3. Stable marriage assignment for man and woman.[18]

Table 1. Key differences and similarities of the general stable marriage problem and taxi dispatch problem.

	General marriage problem[17]	Taxi dispatch problem
Participants	Men and women	Available taxis and new passenger bookings
No. of participants	Equal (N*N)	Not necessarily equal (N*M)
Preference making	Arbitrary – based on personal taste of each man and woman	Passengers prefer a taxi that can pick them up the soonest Taxis prefer passenger requests with the least costs
Preferences complete	Yes, every participant of the matching has ordered preference list of all participants of opposite sex	No, some taxis may be too far away and therefore preference lists may be incomplete
Assignment produced	Once for given number of men and women and their preference lists	Every decision epoch for given number of available taxis and new passenger bookings and their preference lists

The following paragraphs uncover three major factors in the implementation of the stable marriage algorithm for taxis and passengers, which were not yet addresses and propose three solutions to handle them. Firstly, Bai et al.[15] assume in their street hailing study that passengers are willing to wait for infinite amount of time until they are served by a taxi. This study aims to overcome this unrealistic simplification by introducing a constraint, which limits passengers' willingness to wait. If the limit is exhausted, the passenger will seek an alternative transport option. The introduction of the willingness to wait, however, has an implication for the original stable marriage algorithm, which assumes that everyone has a preference about every one of the opposite party. If the taxi is not able to pick up a passenger within the pickup time window, the passenger in fact could not have a preference. This leads to a stable marriage problem with incomplete preference lists that was described by Brito and Meseguer[19]. The preferences of taxis and passengers are constituted in the following fashion: passengers prefer a taxi that can pick them up first; taxi drivers, on the other hand, prefer booking requests with the least costs.

Next, generating assignments in regular time intervals from buffered requests might cause incoherence in the assigned pairs of taxis and passengers from one interval to another. For illustration, imagine a taxi on its way to the passenger with confirmed booking, when a new passenger request appears closer to the current positon of the taxi and the taxi is re-assigned to pick up the closer one. Undoubtedly, taxi drivers would prefer to break the commitment and to pick up the closer one, but with no taxi left to pick up the formerly confirmed passenger, passengers would be frustrated. The remedy is to ensure that the confirmed trips are really picked up and to forbid taxis with 'On call' status to participate on matching process.

The last concern, mentioned but not addressed by Bai et al.[15], relates to the length of the decision epoch. The ultimate aim is to balance two conflicting interests: on one hand to maximize the number of buffered requests for simultaneous assignment and on the other hand to confirm the bookings to the passenger as fast as possible. The longer the buffering interval, the more taxis and passengers happen to be at the same time next to each other, the higher is the potential for improvements. There are theoretical and practical limits to set the decision epoch length. Firstly, if the length of the decision epoch is higher than the willingness to wait, the passenger cannot even get the answer to the booking request. A second issue with utmost importance from the practical user perspective is that any buffering strategy holds the requests before it distributes back the booking confirmations to customers. This implies that the passenger has to wait for the first reply (presumably longer time compared to FCFS strategy). On the other hand, the system response time is known (defined by the length of the decision epoch) and the total waiting time from the moment of booking until the arrival of the taxi could be shorter. However, this benefit is not seen at the first moment and people may be keen on almost instant reply. Therefore, with respect to this rationale, this study deduces that the appropriate length of the decision epoch should not be more than one minute.

4. Simulation model

This section describes the custom-made simulation model and the experiment set-up. The simulation model was developed by the authors using MATLAB and its features are explained below and illustrated in Fig. 4. The simulation model is organized as a discrete event-based simulation. The model schedules events (tasks) and executes them at predetermined points in time. Some events are known and scheduled in advance and others are scheduled only during runtime.

During the initialization phase, the initial positions of taxis, the origins and destinations of booking requests, booking times and desired pickup intervals are generated according to a specified number of taxis, passenger booking requests and a specified distribution. These events are scheduled to be executed at the predetermined time. All the key processes are visualized in Fig. 4.

During runtime, new events are scheduled and scheduled events are executed. For example, if a new passenger request comes in (the scheduled event ‘New booking request’ is executed at a given time), the status of the booking request is changed to ‘Booked’. The event ‘Taxi schedule calculation’ represents the taxi dispatching in which the new booking requests are either confirmed or rejected. Once a booking request has been confirmed, the following events, ‘Passenger begins waiting’ and event ‘Dispatch taxi to trip’ are scheduled. Once the event ‘Dispatch taxi to trip’ has been executed, the status of the taxi changes to ‘On call’ and the event ‘Taxi pick-up passenger’ is scheduled. Upon execution of a ‘Taxi pick-up passenger’ event, the status of the respective taxi changes to ‘Occupied’ and the respective passenger status changes to ‘On trip’. Similarly, upon execution of a ‘Taxi drop-off passenger’ event, the status of taxi changes to ‘Vacant’ and the passenger’s status changes to ‘Delivered’. The simulation ends when there are no more events waiting to be executed.

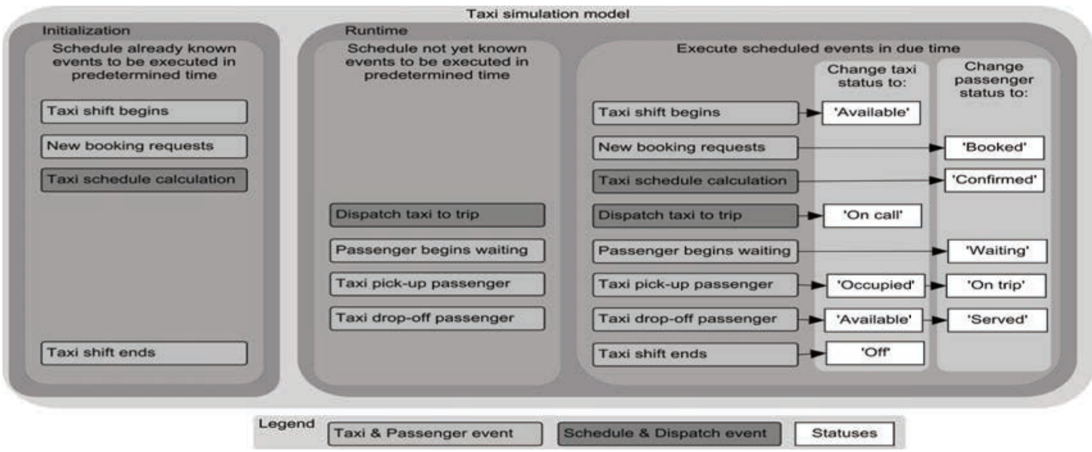

Fig. 4. Taxi simulation model features and structure.

The results are assessed using the following performance indicators: (a) average taxi profit, (b) number of served trips, (c) average vacant distance, (d) average total distance and (e) average passenger waiting time. Profit (P) consists of revenue (R) and cost (C) aspects. Both revenues and costs are triggered either by distance, time or by an event and increase proportionally with the distance, time duration and number of events. Revenue rates are determined by the fare per distance (RD) and per time (RT) and fare surcharge (RE) (e.g. booking fees, position fees). The cost rates are determined by costs of fuel to drive a distance (CD), costs of taxi active time (CT) (taxi driver’s time and vehicle rental per time unit where applicable) and costs of events (CE) (e.g. city center congestion charge and tolls.) The average vacant distance is a sum of the distances driven to the passengers when the status of the taxi was ‘On-call’. This indicator shows the influence of simultaneous assignments on reducing the on-call distance and related costs. The average total distance is a sum of distances driven by the taxis when the taxi was in service (the status is anything except ‘Off’). The total number of pickups, or in other words, the total number of served passengers, reflects the capability of the dispatching strategy to deliver the service. Passenger waiting time indicates the time needed from the desired pickup time (TR) to the actual pickup time (APT). Typically, this is the most critical parameter from the perspective of the passenger.

5. Experiment setup

The introduced factors, described in the methodological section, create new variants of the original dispatching strategy. The stable marriage strategy varies with subsequent addition of the three above factors, passengers’ willingness to wait, commitment to already confirmed requests and appropriate length of decision epoch, which are referred as SM0, SM1, SM2 and SM3 respectively and are summarized in Table 2. The first-come, first-served strategy (FCFS) serves as a common-ground benchmark. Although this is not the most elaborate strategy, it is the most common one in the taxi industry, and is often used for benchmarking in taxi research and, unlike other methods, is relatively easy to replicate.

Table 2. Stable marriage algorithm dispatching strategies variations.

	Dispatching strategy	FCFS	SM0	SM1	SM2	SM3
Factor	Limited passengers’ willingness to wait					
	Commitment to confirmed trips					
	Appropriate length of decision epoch	n.a.				

Booking requests are not known to the model until they are made and the assignment of taxis to passengers is computed in real time. All the booking requests are immediate, meaning that the passengers would like to get the taxi as soon as possible from the time they booked. The taxi fleet size remains unchanged during the simulation. Taxis do not break down. Taxis wait at their initial locations or drop off location of the last served passenger for their next assignment. Taxis do not use any form of anticipation of future requests. The network is Euclidean space, in which taxis can move freely and pick-up and drop-off possible anywhere within this space. The travelling velocity is constant. The physical network, links, turn restrictions, variable travel times and many other features are not considered. While these assumptions do not reflect the reality, they are suitable for drawing initial conclusions and can be extended in the future.

The input parameters used in this study are summarized in Table 3. In order to minimize the influence of the initial set-up on the results, all strategies were simulated in 15 simulation rounds. These 15 spatial-temporal configurations determine the initial taxi positions, origins and destination of the trips and request times. The presented results are based on the mean of these experiments.

Table 3. Input parameters for taxis, passenger trips and the environment.

Abbreviation	Parameter	Value	Unit
n(Taxis, Passengers, Simulations)	Number of taxis, passengers, simulations for each strategy	100, 1200, 15	[1]
Lx, Ly, Ox, Oy, Dx, Dy	Taxi positions and passenger origin and destination positions	Uniformly distributed	[km]
TR	Time of desired pickup time	Uniformly distributed	[hh:mm:ss]
RD, CD	Revenue rate from distance, costs rate of distance	1.560, 0.071	[$/km]
RT, CT	Revenue rate from time, costs rate of time	19.560, 8.300	[$/h]
LD, WW	Length of decision epoch, willingness to wait	30, 1000	[s]
RTI	Time interval when passenger requests can come	4	[h]
SS	Size of square area	20	[km]
TV	Aerial travel velocity	36	[km/h]

6. Results

Results of 15 simulation rounds showing the comparison of FCFS and SM0 through to SM3 (stable marriage based strategies with adding proposed factors) are presented and discussed in this section. The mean and relative improvement over FCFS strategy are summarized in Table 4.

FCFS strategy serves on average 1064 passengers (rounded to integers) out of 1200 (89 %) in less than 13 minutes of the average passenger waiting time. In the perspective of rather high trip demand (three passenger trip requests per hour per taxi), the FCFS strategy sets a high benchmark.

The SM0 does not limit the willingness to wait, which results in both advantages and disadvantages. On one hand, this strategy serves all 1200 passenger trips (100 %) and produces the highest taxi profit among all compared strategies (see Table 4). On the other hand, passengers have to wait almost 20 minutes on average to get a taxi, which might not be acceptable to many passengers.

The SM1 strategy limits willingness to wait. It manages to serve 1027 passengers (86 %.) The average waiting time and total distance are the best among all the compared strategies. Unfortunately, because already confirmed assignments could be changed, the number of confirmed but not picked up passengers reached 26 (3 %.)

The SM2 strategy forbids the changing of confirmed assignment. Indeed, all passengers with confirmed bookings are picked up. However, this is compensated with lower taxi driver profit and increased not-occupied and total distance. This strategy serves 1033 trips (86 %).

The SM3 strategy, thanks to the appropriate length of decision epoch, serves 1089 trips (90 %). As compared to FCFS the number of served trips increases by 2 %, profit increases by 4 %, the highest among the strategies with limited willingness to wait. All other observed parameters also improved: not-occupied and total taxi distance and passengers' waiting time by 19 %, 3 % and 32 % respectively.

Table 4. Mean of performance indicators and relative improvement over FCFS.

		FCFS	SM0	SM1	SM2	SM3	Unit
Mean	Average profit per taxi	93.5	107.9	90.4	90.5	97.6	[$]
	Number of served trips	1063.8	1200.0	1027.4	1033.2	1089.4	[1]
	Average vacant distance	37.5	20.5	21.1	22.3	30.5	[km]
	Average total distance	147.5	145.5	127.5	129.0	143.2	[km]
	Average passenger waiting time	00:12:40	00:20:00	00:08:27	00:08:53	00:08:36	[hh:mm:ss]
Improvement	Average profit per taxi		15 %	-3 %	-3 %	4 %	
	Number of served trips		13 %	-3 %	-3 %	2 %	
	Average vacant distance		45 %	44 %	41 %	19 %	
	Average total distance		1 %	14 %	13 %	3 %	
	Average passenger waiting time		-58 %	33 %	30 %	32 %	

7. Discussion

What does the application of the proposed algorithm mean for the passengers and taxi drivers? Beside the advantages and performance improvements are there any trade-offs? From the passenger's perspective, a dispatching strategy based on concurrent assignment prolongs the initial reply time to either accept or reject the booking. Although the waiting time for this initial reply may be longer, the total waiting time from the request to the pickup is shorter.

Moreover, a relatively high spatial-temporal density of the demand in the simulation experiments, on average three requests per taxi per hour, allows for the effective buffering of the requests and, therefore, highlights the advantages of the methodology. But what happens when the demand is weak, for example during night time hours? Intuitively, with lower demand density, the results will get closer to the FCFS.

8. Conclusion and future work

The proposed dispatching strategy improves the performance of taxi operations in comparison to the first-come, first-served method. The main advantage stems from the stable marriage based simultaneous assignment of new booking requests to available taxis, which improves on-call and total distance, costs and passengers' waiting time.

Benchmarking the performance of the stable assignment with other assignment approaches would be worthwhile. The simulation model can be improved by implementing real network topology with streets, turn and stopping restrictions and stochastic link travel times, cancellation of booking requests and vehicle breakdowns or slowdowns and by relaxing some assumptions (such as non-homogenous vehicle fleet, individualized willingness to wait in relation to the length of the trip) and adding extra features (such as preferences vehicles or workload-balancing mechanisms for drivers, multiple fleet operators). Ideally, two benchmarks would be beneficial: One simple, similar to the one this study uses and one elaborated one, perhaps based on the work by Maciejewski and Bischoff.[11]

Acknowledgements

This study was financially supported by the Singapore National Research Foundation under its Campus for Research Excellence And Technological Enterprise (CREATE) programme. The views expressed herein are solely the responsibility of the authors and do not necessarily represent the official views of the Foundation. The authors would like to express their sincere thanks to all who contributed to this paper.

References

1. Dantzing GB, Ramser JH. The Truck Dispatching Problem. Management Science. 1959;6(1):80-91. doi:10.1287/mnsc.6.1.80.
2. Berbeglia G, Cordeau J-F, Laporte G. Dynamic pickup and delivery problems. European Journal of Operational Research. 2010;202(1):8-15. doi:10.1016/j.ejor.2009.04.024.
3. Angelelli E, Bianchessi N, Mansini R, Speranza MG. Short Term Strategies for a Dynamic Multi-Period Routing Problem. Transportation Research Part C: Emerging Technologies. 2009;17(2):106-119. doi:10.1016/j.trc.2008.02.001.
4. Bailey, Jr WA, Clark, Jr TD. A Simulation Analysis of Demand and Fleet Size effects on taxicab service rates. In: Winter Simulation Conference.; 1987:838-844.
5. Jianxin Y, Xiaomin Z, Hongyu Z. Design and implementation of taxi calling and dispatching system based on GPS mobile phone. Proccedings of the 4th International Conference on Computer Science & Education. 2009. doi:10.1109/ICCSE.2009.5228489.
6. Chang S, Wu C. Comparison of Environmental Benefits Between Satellite Scheduled Dispatching and Cruising Taxi Services. Proceedings of the Transportation Research Board 89th Annual Meeting. 2010;(1).
7. Grau JMS, Romeu MAE. Agent Based Modelling for Simulating Taxi Services. Procedia Computer Science. 2015;52:902-907. doi:10.1016/j.procs.2015.05.162.
8. Lee D-H, Wu X. Dispatching strategies for the taxi-customer searching problem in the booking taxi service. In: Proceedings of the Transportation Research Board 92nd Annual Meeting.; 2013.
9. Lee D-H, Wang H, Cheu R, Teo S. A taxi dispatch system based on current demands and real-time traffic conditions. Transportation Research Record. 2004;1882(1):193-200. doi:10.3141/1882-23.
10. Ngo MN. Fuzzy linear assignment problem: an approach to vehicle fleet deployment. In: Proceedings of the IEEE International Conference on Fuzzy Systems.Vol 2.; 2004. doi:10.1109/FUZZY.2004.1375583.
11. Maciejewski M, Bischoff J. Large-scale microscopic simulation of taxi services. Procedia Computer Science. 2015;52(Ant):358-364. doi:10.1016/j.procs.2015.05.107.
12. Pureza V, Laporte G. Waiting and buffering strategies for the dynamic pickup and delivery problem with time windows. INFOR: Information Systems and Operational Research. 2008;46(3):165-175. doi:10.3138/infor.46.3.165.
13. Seow KT, Sim K. Collaborative assignment using belief-desire-intention agent modeling and negotiation with speedup strategies. Information Sciences. 2008;178(4):1110-1132. doi:10.1016/j.ins.2007.09.024.
14. Seow KT, Lee DH. Performance of multiagent taxi dispatch on extended-runtime taxi availability: A simulation study. IEEE Transactions on Intelligent Transportation Systems. 2010;11(1):231-236. doi:10.1109/TITS.2009.2033128.
15. Bai R, Li J, Atkin JA, Kendall G. A novel approach to independent taxi scheduling problem based on stable matching. Journal of the Operational Research Society. 2014;65(10):1501-1510. doi:10.1057/jors.2013.96.
16. Shamma J. Task assignment for mobile agents. In: Cooperative Control of Multi-Agent Systems.; 2008:109-137.
17. Gale D, Shapley L. College admissions and the stability of marriage. American Mathematical Monthly. 1962;09(1):9-15.
18. McVitie D, Wilson L. The stable marriage problem. Association for Computing Machinery (ACM). 1971;14(7):486-490.
19. Brito I, Meseguer P. Distributed stable marriage with ties and incomplete lists. In: Proceedings of Principles and Practice of Constraint Programming - CP 2006.; 2006. doi:10.1007/11889205_49.