Dispatching Strategies for the Taxi-Customer Searching Problem in the Booking Taxi Service

Der-Horng Lee
(Corresponding author)
Department of Civil & Environmental Engineering
National University of Singapore
Block E1A, #07-16, 1 Engineering Drive 2
Singapore, 117576
Phone: (+65)65162131
Email: dhl@nus.edu.sg

Xian Wu
Department of Civil & Environmental Engineering
National University of Singapore
Block E1, #08-20, 1 Engineering Drive 2
Singapore, 117576

Submission Date: Nov 15, 2012
Word Count: 6173 words + 4 figures + 1 tables = 7423
ABSTRACT

Automatic taxi dispatching has been widely used in many large cities worldwide, in which customers can book taxis through phones or mobile devices. Two types of bookings are commonly known: one is the Current Booking (CBK), the customer makes a booking call for a taxi that can reach him/her as early as possible; another is the Advance Booking (ABK), the customer makes a booking call and indicates the pickup time which is normally in half an hour or later. In this paper, taking the taxi by making either CBK or ABK is defined as the Booking Taxi Service (BTS) while taking the taxi by either waiting at taxi stand or hailing on the street is defined as the Non-Booking Taxi Service (NBTS). In order to evaluate different dispatching strategies, a Taxi-Customer Searching Problem (TCSP) is formulated in this paper in which both BTS and NBTS are considered, and different dispatching strategies are evaluated and compared. The microscopic traffic simulation is adopted as the approach for modeling and analysis of the TCSP. A sensitivity analysis by varying the booking demand is conducted based on the simulation, and the simulation results show that the Advance Booking Chain Dispatching Strategy (ABC-DS) can give better operational performance in certain demand levels which may have the potential to attract more customers to take the taxi by booking in advance. Moreover, these results can also provide strategic implications for the taxi operators.

Keywords: Advance Booking Chain Dispatching Strategy (ABC-DS), Booking Taxi Service (BTS), Microscopic simulation model, Taxi-Customer Searching Problem (TCSP)
1. INTRODUCTION

One problem existed in the current taxi service market is the imbalance between taxi supply and demand (1), which may cause two negative impacts on the demand side and supply side of the taxi service market: one is the longer waiting time of customers who are waiting at the taxi stand or on the street; the other is the longer empty cruising time of taxis. These two negative impacts cause not only the waste of social resources for both customers and taxi drivers but also environmental problems such as the emissions released from taxis when they are searching and waiting for customers on the congested road network.

In response to the aforementioned problem, automatic taxi dispatching approaches have been widely used in many large cities worldwide, in which customers can book taxis directly through phones or mobile devices and the taxi operator may employ the taxi dispatching system to deal with the customer bookings (2, 3). Compared with the traditional ways of taking taxis by hailing on the street or waiting at the taxi stand, booking taxis through the dispatching system has more advantages: it provides an alternative way for the customer and the taxi driver to find each other easily. Two types of bookings have been commonly known: one is the Current Booking (CBK): the customer makes a booking call for a taxi that can reach him/her as early as possible; another is the Advance Booking (ABK): the customer makes a booking call and indicates the pickup time which is at least in half an hour later (4).

To differentiate between the taxi service with bookings to that without bookings, the following two terms are defined in this paper:

- Booking Taxi Service (BTS): provided for the customer who takes the taxi by booking (either CBK or ABK) through the taxi dispatching system;
- Non-Booking Taxi Service (NBTS): provided for the customer who takes the taxi by either waiting at the taxi stand or hailing on the street.

Many research studies on the topic of taxi service can be found in the literature (4-14), but only a few of them focused on the dispatching approaches for the BTS (either CBK or ABK). For example, for the CBK, Both Lee et al. (11) and Seow and Lee (13) explored efficient dispatching strategies for it: the former presented a shortest travel time based dispatching strategy while the latter proposed an agent-based dispatching strategy which enabled taxis to negotiate and cooperate with each other to achieve a group objective; for the ABK, Lee et al. (4) applied the Pickup and Delivery Problem with Time Windows (PDPTW) approach to deal with the ABK, in which a series of ABKs would be automatically chained and then sent to individual taxis in the form of booking packages. Notice that the aforementioned research studies employed the microscopic traffic simulation approach to model and test the dispatching strategies.

However, there are two limitations existed in the previous research studies on the dispatching approaches. The first limitation is that those studies modeled either CBK or ABK (and the respective dispatching strategies), but neither modeled them together nor proposed a dispatching approach to handle them concurrently. This limitation may cause the inadequateness of those studies to evaluate the performance of dispatching strategies: on one hand, both the CBK and the ABK may become constraints for the dispatching processes of each other, e.g., a taxi with an already confirmed ABK will not decide to accept a new CBK if the delivery of the new CBK may cause the delay of delivering the ABK, so that the overall dispatching performance may be over-estimated without considering these constraints; on the other hand, dealing only with the CBK or only with the ABK may under-estimate the overall dispatching performance, since the CBK and the ABK could be scheduled together to achieve
better overall dispatching performance.

The second limitation is that the previous studies yet modeled the NBTS, which may also cause the inadequateness of those studies to evaluate the performance of dispatching strategies: on one hand, same as the case of CBK, the NBTS may become the constraint for the dispatching process of the ABK, e.g., a taxi with an already confirmed ABK will not pick up a customer waiting at a taxi stand or on the street if the delivery of the customer may cause the delay of delivering the ABK, so that the overall dispatching performance may be over-estimated without considering this constraint; on the other hand, the NBTS will directly affect the demand for BTS, e.g., the customer will choose NBTS rather than BTS if the supply of NBTS is sufficient or the supply of BTS is insufficient, which may also indirectly affect the overall dispatching performance.

Thus, this paper will further strengthen the previous studies and has the research objectives listed in the following:

1. Model the taxi service market with both BTS (CBK and ABK) and NBTS;
2. Model and evaluate a dispatching strategy which is similar to the one of the real world, namely the Separate Dispatching Strategy (Sep-DS): bookings (CBK or ABK) will be assigned to taxis separately in which the dispatching process only needs to ensure that each booking assignment will not conflict with others;
3. Model and evaluate an improved dispatching strategy, namely the Advance Booking Chain Dispatching Strategy (ABC-DS): ABKs are allowed to be chained up and scheduled in the dispatching process;
4. Examine the application ranges of the two dispatching strategies (Sep-DS and ABC-DS) in terms of different demand levels, and explore the effects of NBTS to the dispatching.

The problem to be studied in this paper can be described as the Taxi-Customer Searching Problem (TCSP) in which both BTS and NBTS are considered, and different dispatching strategies will be evaluated and compared. The microscopic traffic simulation is adopted as the approach for modeling and analysis. A programming plugin using the APIs (Application Programming Interfaces) of the simulator is designed to enable the simulation of dynamic customer behaviors and the dispatching strategies. Various booking demand levels will be tested, and the performance of dispatching strategies will be evaluated by three performance indicators: 1) the taxi Occupancy Rate (OR): the ratio between the total occupied time and the total operating time of all taxis; 2) the Customer Waiting Time (CWT): the average waiting time of all customers; and 3) the numbers of completed and unsuccessful bookings. A Wasp-like Agent (WA) algorithm is proposed in this paper to solve the scheduling problem in ABC-DS.

The rest of the paper will be organized as follows: the problem formulation will be presented in Section 2; the development of the simulation model and the WA algorithm will be introduced in Section 3; the simulation results and analysis will be presented in Section 4; followed by the conclusion in Section 5.

2. PROBLEM FORMULATION

In the Taxi-Customer Searching Problem (TCSP), taxis are assumed to be running on a road network $G = (V, E)$, $V$ is the set of nodes and $E$ is the set of links. A taxi $T_i$ can pick up or drop customer(s) at a taxi stand $TS_i \in TS \subset V$ where $TS$ is the set of all taxi stands (for simplification but without loss of generality, the case of picking up or dropping customers on a road segment $V_i \in V$ will not be considered in this paper). The objective of the problem is to test and compare the aforementioned two dispatching strategies (Sep-DS and ABC-DS).
2.1. Assumptions for Customer Behaviors

2.1.1. The Generation of Customers
Customers are assumed to be generated by a stochastic process over time $t$, which is modeled as a Poisson point process that the time interval between the generations of each pair of consecutive customers with the same Origination-Destination (OD) pair has an exponential distribution with parameter $\lambda$, and these time intervals are assumed to be independent with each other. Thus, the number of customers with the same OD generated in time interval $(t, t + \Delta t]$ follows the Poisson distribution with the parameter $\lambda\Delta t$, so that:

$$\Pr[N(t + \Delta t) - N(t) = k] = \frac{e^{-\lambda\Delta t} (\lambda\Delta t)^k}{k!}, \quad k = 0, 1, \ldots,$$

where $N(t + \Delta t) - N(t) = k$ is the number of customers (with the same OD) that are generated in time interval $(t, t + \Delta t]$. In this problem a “customer” refers to one person or a group of persons who are willing to take one taxi together. Once a customer is generated, he/she may choose to either queue at a taxi stand or make an ABK.

2.1.2. Customers Queuing at Taxi Stands
Customers who have chosen to queue will be queuing and waiting at the taxi stands for the NBTS, i.e., the customer queuing in the very front of the queue will take the next available taxi arrived to the taxi stand. If a customer has been waiting for a certain period of time $T_0$ but no taxi picks up him/her, the customer may start to make a CBK to the dispatching system. Denote $C_j$ as the $j^{th}$ generated customer in the problem and $CBK_j$ is the CBK made by $C_j$, then $CBK_j$ should indicate the pickup location denoted as $p_j \in V$ and the delivery location denoted as $p_j \in V$ of the customer. There are three possible consequences after $CBK_j$ is made:

- If a taxi arrives to the taxi stand and ready to pick up the customer $C_j$ before $CBK_j$ is confirmed by any other taxi, $C_j$ will board on the taxi and then cancel $CBK_j$;
- If a taxi arrives to the taxi stand and ready to pick up the customer $C_j$ but $CBK_j$ is already confirmed by another taxi $T_i$, $C_j$ will not board on the arrived taxi and will continue to wait for the arrival of taxi $T_i$;
- If the customer $C_j$ has been waiting for a certain period of time $T_{\text{max}}$, $C_j$ will decide not to wait any longer and $CBK_j$ will be treated as an unsuccessful booking.

2.1.3. Customers Making the ABK
Customers who have chosen to make the ABK will just make the bookings to the dispatching system after they are generated. Denote $ABK_j$ as the ABK made by the $j^{th}$ generated customer $C_j$. Besides the pickup and delivery locations $p_j^\pi$ and $p_j^\delta$, $ABK_j$ also has to indicate the Desired Pickup Time (DPT) denoted as $t_j^{\text{DPT}}$ and Estimated Delivery Time (EDT) denoted as $t_j^{\text{EDT}}$ of the customer. The DPT is at least 30 minutes later than the generating time of the ABK and following a probability distribution (e.g. normal distribution) with a pre-specified standard deviation (e.g. 30 minutes). If the customer $C_j$ has been waiting for a certain period of time $T_{\text{max}}$ after the DPT, $ABK_j$ should be treated as an unsuccessful booking.

2.2. Assumptions for Taxi Operations

2.2.1. Operational States of the Taxi
Taxis are assumed to be homogeneous agents with same kinematics properties. They are running randomly on the road network between taxi stands where they are queuing for
boarding customers (This simplified assumption is due to that this paper focuses on the
evaluation of dispatching strategies in a short-term time period and in a small study area, e.g.,
the peak-hours of a typical working day in the central area of a city). Five operational states
are defined for the taxi: 1) Free state: the taxi is running on the road network, no customer is
occupied and no booking request is assigned to the taxi; 2) On-call state: the taxi is running
on the road network, no customer is occupied but there is one booking request assigned to the
taxi, and the taxi is heading for the taxi stand where the booking request comes from; 3)
Boarding state: the taxi is boarding a customer at the taxi stand, the customer may or may not
have placed a booking request; 4) Boarded state: the taxi is running on the road with a
customer occupied, and the taxi is heading for the destination of the customer; 5) Alighting
state: the taxi is alighting a customer at the taxi stand.

2.2.2. Decision Process of the Taxi for Dealing with Bookings
The taxi (agent) is assumed to be attached with two data structures which aim to facilitate the
decision process of the taxi in dealing with the bookings (CBK or ABK) assigned to it by the
dispatching system:

- Data structure 1: A pointer pointing to the CBK which is already confirmed by the taxi;
  the pointer will be set to null if the taxi has no confirmed CBK;
- Data structure 2: A pointer pointing to the Advance Booking Queue (ABK-Q) which is a
collection of ABKs already confirmed by the taxi; the ABKs in the ABK-Q are sorted by
  their DPTs (early to later).

The taxi should ensure that the following four rules are satisfied during the entire
operating period of the taxi (also shown in Figure 1):

```
Rules for the decision process of the taxi.
```

![Diagram of taxi decision process]

Notes:
BK: Booking
CBK: Current Booking
ABK: Advance Booking
ABK-Q: Advance Booking Queue
DPT: Desired Pickup Time

Legends:
- Pointer
- Assign
- No BK confirmed
- BK confirmed

FIGURE 1 Rules for the decision process of the taxi.
1) The pointer pointing to the CBK and the pointer pointing to the ABK-Q are attached to the taxi throughout the duration of operation;
2) If the taxi has at least one ABK in its ABK-Q, the taxi could accept a new CBK only when the delivery of the new CBK will not cause the delay of delivering the existing ABKs in the ABK-Q;
3) If the taxi is serving a CBK (the taxi is in On-call state) or occupied with a customer, it could accept a new ABK only when the delivery of the currently serviced CBK will not cause the delay of delivering the new ABK, and the delivery of the new ABK will not cause the delay of delivering the existing ABKs in the ABK-Q;
4) The taxi will check from time to time to ensure that when the current time is approaching the DPT of any ABK in the ABK-Q, the taxi could assign the ABK (from the ABK-Q) to the CBK pointer and start to head to the pickup location of it.

2.3. The Dispatching Strategies

2.3.1. Separate Dispatching Strategy (Sep-DS)
The Sep-DS is the strategy that all bookings will be tried to assign to available taxis separately, and an assignment can be regarded as successful if a taxi confirms to take it; otherwise, the assignment will be regarded as unsuccessful and the booking will be re-assigned later. For the CBK, the dispatching is based on the shortest travel time rule (11), in which the CBK will be assigned to the available taxi that can reach the customer with the shortest travel time. For the ABK, any available taxi could receive the ABK from the dispatching system, but the confirmation needs the prior satisfaction of the rules mentioned in Section 2.2.2. In Sep-DS, the maximum length of the ABK-Q of each taxi is 1, and no specific algorithm is designed for it due to the simplification of the strategy.

2.3.2. Advance Booking Chain Dispatching Strategy (ABC-DS)
The ABC-DS is the strategy that enables each taxi to form an Advance Booking Chain (ABC) based on its existing ABK-Q structure. There is no maximum length requirement for the ABK-Q of each taxi in ABC-DS, and the taxi dispatching for the CBK in this strategy is the same with that in Sep-DS.

Definition 1 Advance Booking Chain (ABC): the ABC of the taxi $T_i$ can be denoted as $ABC_i$, which is a set of sequenced ABKs such that $ABC_i = \{ ABK_1^i, ..., ABK_{N_i}^i \}$ where $|ABC_i| = N_i$. Denote $ABK_m^i$ as the $m^{th}$ ABK in $ABC_i$, and $p_{i,m}^+, p_{i,m}^-, t_{i,m}^{DPT}$ and $t_{i,m}^{EDT}$ are the pickup location, delivery location, DPT and EDT of $ABK_m^i$. An ABC should satisfy the following rules:

- All the pickup and delivery locations $p_{i,m}^+$ and $p_{i,m}^-$ ($m=1,2,\ldots,N_i$) should be visited by $T_i$;
- $p_{i,m}^+$ and $p_{i,m}^-$ ($m=1,2,\ldots,N_i$) should be directly connected in the route traveled by $T_i$, and $p_{i,m}^+$ is visited before $p_{i,m}^-$;
- $T_i$ should arrive at any location $p_{i,m}^+$ ($m=1,2,\ldots,N_i$) within the time interval $[t_{i,m}^{DPT} - \varepsilon, t_{i,m}^{DPT} + \varepsilon]$ where $\varepsilon$ is a pre-specified random error of $t_{i,m}^{DPT}$. If $T_i$ arrives to $p_{i,m}^+$ earlier than $t_{i,m}^{DPT} - \varepsilon$, the taxi needs to wait until time $t_{i,m}^{DPT} - \varepsilon$;
- $ABC_i$ should be scheduled whenever a change of the ABKs in $ABC_i$ occurs, which is expected to achieve a better operational performance of the taxi services.

The idea of ABC-DS is from Lee et al. (4), however, the problem conditions in this
paper is more complicated than those in Lee et al. (4), which is because: 1) this paper has considered the modeling of both BTS (CBK and ABK) and NBTS; 2) the ABK considered in this paper is on-demand, i.e., ABKs will be generated randomly over time. Hence, a Wasp-like Agent (WA) algorithm (16-20) is introduced in the next section to solve the scheduling problem in ABC-DS.

3. METHODOLOGY

3.1. The Microscopic Traffic Simulation

The microscopic traffic simulation is adopted as the modeling and analysis approach for the TCSP introduced in Section 2. It includes a microscopic traffic simulation software - PARAMICS (21) and a plugin designed by programming with the Application Program Interfaces (APIs) which enable the software to simulate the customer dynamic behaviors and the taxi operations as well as the the dispatching strategies.

In the simulation model, the central region of Singapore is chosen as the study area which covers an area of around 17km × 10.5km. The central region of Singapore and the study network are both shown in Figure 2. There are totally 993 nodes and 2,348 links in the network, and 193 zones defined to represent the taxi stands.

![Figure 2: The central region of Singapore and the road network in the simulation.](image)

3.2. The Wasp-like Agent Algorithm for the Scheduling Problem in ABC-DS

A Wasp-like Agent (WA) algorithm is developed to solve the scheduling problem in ABC-DS. The algorithm will be performed for each new ABK immediately after the ABK is generated. Denote $ABK_j$ as the ABK made by the $j$th generated customer, the pseudo code for the WA algorithm is shown in Table 1.

<table>
<thead>
<tr>
<th>TABLE 1 The Pseudo Code for the Wasp-like Agent (WA) Algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Input:</strong> set of taxis $T$, $ABK_j$ generated at time $t$</td>
</tr>
<tr>
<td><strong>Output:</strong> $ABK_j$ assigned to a taxi in $T$, and necessary re-scheduling of ABCs for some taxis in $T$</td>
</tr>
<tr>
<td>1 Phase 1: The initial assignment phase</td>
</tr>
<tr>
<td>2 Empty $CT$ which is the set of candidate taxis for the assignment of $ABK_j$</td>
</tr>
<tr>
<td>3 For Each taxi $T_i \in T$</td>
</tr>
<tr>
<td>4 $T_i$ calculates $S_{ij} = S(T_i, ABK_j)$ which is the stimulus of $ABK_j$ to $T_i$</td>
</tr>
<tr>
<td>5 $T_i$ calculates $\theta_{ij} = \theta(T_i, ABK_j)$ which is the response of $T_i$ to $ABK_j$</td>
</tr>
<tr>
<td>6 If $\theta(T_i, ABK_j)$ returns null</td>
</tr>
<tr>
<td>7 Continue;</td>
</tr>
<tr>
<td>8 Else</td>
</tr>
<tr>
<td>9 $T_i$ calculates $P_j = S_{ij}^2 / (S_{ij}^2 + \theta_{ij}^2)$</td>
</tr>
<tr>
<td>10 Insert $T_i$ to $CT$</td>
</tr>
</tbody>
</table>
There are two phases in the algorithm: one is the initial assignment phase and the other is the local planning phase. There are also four sub-modules mentioned (but yet detailed) in the pseudo code for the WA algorithm: $S(T_i, ABK_j)$, $\theta(T_i, ABK_j)$, Insert($ABK^i_m$, $ABC_i$) and Swap($ABK^i_m$, $ABC_i$). The two phases and the four sub-modules will be introduced in Section 3.2.1 and 3.2.2.

3.2.1. Phase 1: the Initial Assignment Phase

The objective of the initial assignment phase is to quickly find a feasible location for $ABK_j$, and also to shorten the system response time to the customer. In this phase, the stimulus $S_{ij}$ and the response $\theta_{ij}$ will be calculated respectively.

Definition 2 the stimulus $S_{ij}$: $ABK_j$ has a stimulus $S_{ij}$ to the taxi $T_i$, which could be considered as the benefit the taxi could obtain, e.g., the potential incremental occupied time of the taxi. So that the travel time from $p_j^+$ to $p_j^-$ is chosen to represent the stimulus $S_{ij}$.

$$S_{ij} = S(T_i, ABK_j) = TT(p_j^+, p_j^-)$$

Where the function $TT(p_j^+, p_j^-)$ is the travel time between locations $p_j^+$ and $p_j^-$. 

Definition 3 the response $\theta_{ij}$: $\theta_{ij}$ is the response of the taxi $T_i$ to $ABK_j$. Suppose that there exists a pair of successive ABKs in $ABC_i$ which are denoted as $ABK^i_m$ and $ABK^i_{m+1}$, satisfying that:

$$t_{ij, m}^{DPT} - t_{ij, m}^{EDT} \geq TT(p_{ij, m}^-, p_{ij, m}^+) - \varepsilon \quad (3)$$

$$t_{ij, m+1}^{DPT} - t_{ij, m+1}^{EDT} \geq TT(p_{ij, m+1}^+, p_{ij, m+1}^-) - \varepsilon \quad (4)$$

Where $t_{ij, m}^{DPT}$, $t_{ij, m}^{EDT}$, $p_{ij, m}^+$, $p_{ij, m}^-$ are the DPT, EDT, pickup location and delivery location of $ABK^i_j$, $t_{ij, m+1}^{DPT}$, $t_{ij, m+1}^{EDT}$, $p_{ij, m+1}^-$ are the DPT and delivery location of $ABK^i_{m+1}$, and $\varepsilon$ is a pre-specified random error of DPT. Equations (3)
and (4) are to ensure that at least one feasible location should be existed in ABC for ABKj; otherwise, the taxi Ti will not be considered as the candidate for the assignment of ABKj. Then the response θij can be calculated as shown in Equation (5), which could be interpreted as the incremental cost of the taxi, i.e., the incremental empty cruising time of the taxi for serving ABKj:

$$\theta_{ij} = \theta(T_i, ABK_j) = TT(p_{i,m}^-, p_{j}^+) + TT(p_{j}^+, p_{i,m+1}^-) - TT(p_{i,m}^-, p_{i,m+1}^+)$$

Then, all taxis with feasible locations for ABKj will be inserted to CT which is the set of candidate taxis for the assignment of ABKj, and each taxi Ti∈CT will calculate $P_{ij} = S_{ij}^2/(S_{ij}^2 + \theta_{ij}^2)$ where $P_{ij}$ is the probability that taxi Ti decides to optimize the ABC. ABKj will then be assigned to the taxi in CT with the highest $P_{ij}$, i.e., taxi $T_i^*$ that $P_{i^*,j} = \max_{T_i \in CT} P_{ij}$. If there is no candidate taxi due to the violation of Equations (3) and (4), no assignment will be performed at the moment.

3.2.2. Phase 2: the Local Planning Phase

After the initial assignment phase, a subsequent local planning phase will be performed to further improve the solution. The motivation for this is explained in the following:

- Assume that in the initial assignment phase, $ABK_j$ is assigned to the taxi $T_i$, and the corresponding probability $P_{i,j}$ is the highest one that all current available taxis can provide;
- The initial assignment phase is fast, but it can only provide a feasible but not optimal solution. For example, when another ABK denoted as $ABK_{j'}$ is generated, it will be assigned to the taxi $T_{j'}$ with the highest probability $P_{i',j'}$ that all current available taxis can provide; however, if $(P_{i,j} + P_{i',j'}) < (P_{i,j'} + P_{i',j'})$, it will be better to assign $ABK_j$ to $T_i$ and $ABK_{j'}$ to $T_{j'}$ respectively;
- Because of the dynamic nature of the ABK, the final optimal solution can only be identified after the last ABK is generated; however, at every time $t$ during the taxi operation, there is a current feasible solution that can be improved further;
- Thus, a subsequent local planning phase will be performed to further improve the initial assignment.

Several considerations also need to be paid attention in performing the local planning phase, which are:

- The local planning phase will be performed only when there is still computational time available. If the next ABK is generated quickly after the initial assignment of the current one, there may be no time to further improve it;
- The local planning phase only deals with the ABKs already generated but yet to be served;
- A new round of local planning phase will start after the initial assignment of a newly generated ABK, which will be performed iteratively until either the local planning phase is finished or another ABK is generated.

The local planning phase consists of iterations of move operations. The move operations are to adjust the current assignment of ABKs to search for an improved one. Two move operations will be considered: the first is the insert move and the second is the swap move. If a move operation (insert or swap) results in a better assignment, the move operation will be confirmed and successful; otherwise, the ABKs will remain in their current ABCs.
**Definition 4** insert move: the insert move is to remove an ABK from the ABC of its current taxi and then insert it into the one of another. As shown in Table 1, the sub-module Insert\(\{ABK_{m}^{i}, ABC_{i}\}\) performs an insert move to try to insert \(ABK_{m}^{i} \in ABC_{i}\) between every possible locations in \(ABC_{i}\). For example as shown in Figure 3(a), taxi \(T_{j}\) tries to delete \(ABK_{m}^{i}\) from \(ABC_{i}\) and then insert it into \(ABC_{i'}\) between \(ABK_{m-1}^{i'}\) and \(ABK_{m+1}^{i'}\). The cost reduction \(\Delta c_{\text{insertion}}\) is shown in Equation (6).

\[
\Delta c_{\text{insertion}} = [TT(p_{m-1}^{-}, p_{m}^{+}) + TT(p_{m}^{-}, p_{m+1}^{+}) + TT(p_{n}^{-}, p_{n}^{+})] - \\
[TT(p_{m-1}^{-}, p_{m}^{+}) + TT(p_{m}^{-}, p_{m+1}^{+}) + TT(p_{m-1}^{-}, p_{m+1}^{+})]
\]

**Definition 5** swap move: the swap move is to swap two ABKs from two ABCs of taxis. As shown in Table 1, the sub-module Swap\(\{ABK_{m}^{i}, ABC_{i}\}\) performs a swap move to try to swap \(ABK_{m}^{i} \in ABC_{i}\) with each ABK in \(ABC_{i'}\). For example as shown in Figure 3(b), taxi \(T_{i}\) tries to swap \(ABK_{m}^{i} \in ABC_{i}\) with \(ABK_{m}^{i'} \in ABC_{i'}\). The cost reduction \(\Delta c_{\text{swap}}\) is shown in Equation (7).

\[
\Delta c_{\text{swap}} = [TT(p_{m-1}^{-}, p_{m}^{+}) + TT(p_{m}^{-}, p_{m+1}^{+}) + TT(p_{n}^{-}, p_{n}^{+}) + TT(p_{m}^{+}, p_{n+1}^{+})] - \\
[TT(p_{m-1}^{-}, p_{m}^{+}) + TT(p_{m}^{-}, p_{m+1}^{+}) + TT(p_{m-1}^{-}, p_{m+1}^{+}) + TT(p_{m}^{+}, p_{n}^{+})]
\]

**FIGURE 3** The move operations.

The move operation (insertion or swap) will be treated as unsuccessful if \(\Delta c_{\text{insertion}} < 0\) or \(\Delta c_{\text{swap}} < 0\). The two constraints described in Equations (3) and (4) should also be satisfied in the iterations of move operations; otherwise, the move operation will be treated as unsuccessful. In the local planning phase, a tabu-list is designed to prevent an ABK from being assigned to a taxi twice before a certain number move operations are performed, in which similar data structures are designed by following the approach proposed by Nanny and Barnes (22).
4. SIMULATION EXPERIMENTS

4.1. Simulation Settings
There are totally 500 taxis to be simulated by the model introduced in 3.1. A series of experiments are conducted for 10 demand levels of the customer which are from 800 arrivals/hour to 8,000 arrivals/hour with the increment of 800 arrivals/hour. In order to observe the effect of the ABK to the performance of the dispatching, we arbitrarily assume that 50% of the customers will make the ABK, and the other 50% of the customers will chose NBTS or make the CBK.

Both dispatching strategies, Sep-DS and ABC-DS are simulated for each demand level. The performance of each strategy in each demand level is evaluated in terms of the OR and CWT. In addition, the number of completed bookings and the number of unsuccessful bookings are also recorded in each simulation scenario.

The standard deviation of the DPT of all ABKs is set to 30 minutes. The customer’s start-to-book time $T_0$ is set to 3 minutes, and the maximum time $T_{\text{max}}$ the customer can wait at the taxi stand is set to 10 minutes. The total simulation period is 2 hours plus 30 minutes warm-up time. Other parameters of the simulation model are set based on the field observation and survey.

4.2. Simulation Results and Analysis
The simulation results are shown in Figure 4.

**FIGURE 4** The simulation results for both dispatching strategies.
4.2.1. The Occupancy Rate (OR)

It can be found in Figure 4(a) that the ABC-DS can improve the overall OR up to 6.77% (demand level = 4) compared with Sep-DS when the demand level < 6. This can be explained as follows: firstly, the ABC-DS chains up ABKs which enables the taxi to avoid unnecessary empty cruising time to a certain extent; secondly, the ABC-DS tries to assign the ABK to the taxi with the highest stimulus (the higher occupancy time with the lower empty cruising time), which may directly lead to the increase of the OR of the entire taxi fleet.

However, it is also found in Figure 4(a) that in terms of improving the overall OR, the ABC-DS is no better than the Sep-DS only when the demand level is high (> 6). This can be also explained as follows: when the demand level is high (> 6), most taxis will be too busy to accept the incoming CBKs so that the demand for the NBTS will become higher. In such a situation, a taxi under Sep-DS will have higher chance to offer the NBTS to a new customer at the taxi stand where it has just dropped one; however, a taxi under ABC-DS will has little chance to offer the NBTS, since it has to head for the pickup location of the next ABK emptily after it drops one at a taxi stand, which may cause the increase of the taxi empty cruising time, in other words, cause the decrease of the OR of the entire taxi fleet.

4.2.2. The Customer Waiting Time (CWT)

Figure 4(b) shows that the ABC-DS can shorten the CWT as much as 34.65% (demand level = 3) when the demand level is low (< 6); however, when the demand level is high (> 6), the ABC-DS shows no advantage than the Sep-DS in terms of the CWT.

4.2.3. The Numbers of Completed Bookings and Unsuccessful Bookings

Figure 4(c) shows that the ABC-DS can complete more ABKs but less NBTS+CBK than Sep-DS. This may be due to the reason that the former strategy allows more than one ABK to be assigned to a taxi which enables the taxi to serve more ABKs. Figure 4(d) shows that the total number of unsuccessful bookings in the ABC-DS is less when the demand level < 7, which indicates that the ABC-DS may be a potential way in attracting more customers to take the taxi by booking in advance.

5. CONCLUSIONS AND FUTURE WORKS

This paper has further improved previous studies on the taxi dispatching approaches in the following aspects:

- A TCSP is formulated, in which both the BTS (CBK and ABK) and the NBTS are modeled;
- Two dispatching strategies (Sep-DS and ABC-DS) are simulated, and a WA algorithm is proposed to solve the scheduling problem in ABC-DS. The operational performances and application ranges of the two dispatching strategies are evaluated in terms of the OR, the CWT and other performance indicators. The effects of the NBTS to the dispatching performance are also examined;
- The microscopic traffic simulation is adopted as the approach for modeling and analysis of the TCSP, and a programming plugin based on the APIs of the selected simulator is designed to enable the simulation of dynamic customer behaviors, taxi operations and dispatching strategies.

Two implications can be obtained from the simulation results: on one hand, the ABC-DS is effective only in certain demand levels (< 6), and in this case, the taxi driver will be benefited from the higher OR (more profitable); the customer will also be benefited not
only because of shorter waiting time but also the improved probability of getting a taxi; on
the other hand, the ABC-DS will be ineffective when the demand level is high (> 6). These
implications are useful for the taxi operators to make strategic decisions. For example, the
ABC-DS is highly recommended to the taxi operator who has comparatively low booking
demands, which may apply to those small taxi companies who have lower market shares; for
the taxi operator who has comparatively high booking demands, using the ABC-DS should be
very careful, which may apply to those large taxi companies with dominant market shares.

The future works of this paper include:

- In the simulation experiments, a larger road network (e.g., the entire island of Singapore)
  will be chosen as the study area;
- In the simulation experiments, the ratio of customers who will make the ABK is
  arbitrarily set to 50% in this paper, so the sensitivity analysis by varying this ratio will be
  conducted and analyzed;
- In the simulation experiments, the sensitivity analysis by changing the customer demand
  scenarios will be conducted and analyzed;
- In the modeling of the taxi service in a larger road network, the taxi’s dynamic searching
  behaviors (e.g., the choice of destination for picking up passengers) will be considered.

REFERENCE

   Dispatch System. Presented at the 6th International Conference on Mobile Systems,
   Applications, and Services, Breckenridge, Colorado, USA, 2008
2. Xu, Z. C., Y. F. Yuan, H. L. Jin and H. Ling. Investigating the Value of Location
   Information in Taxi Dispatching Services: A case study of DaZhong Taxi. In the
   Proceedings of the 9th Pacific Asia Conference on Information Systems, Bangkok,
   Thailand, 2005
3. Liao, Z. Q. Taxi dispatching via global positioning systems. *IEEE Transactions on
   Strategy to Reduce Cost of Taxi Operations. Presented at the 83th Annual Meeting of the
   Transportation Research Board, Washington, D.C., 2004
6. Aquilina, M. Quantity De-restriction in the Taxi Market Results from English Case
   607-616
10. Chang, S. K. J. and C. H. Chu. Taxi Vacancy Rate, Fare, and Subsidy with Maximum
    Social Willingness-to-Pay Under Log-Linear Demand Function. In *Transportation
    Research Record: Journal of the Transportation Research Board*, No. 2111,


