A probabilistic model for vehicle scheduling based on stochastic trip times

Yindong Shen [a,b,*], Jia Xu [a,b], Jingpeng Li [c]

[a] School of Automation, Huazhong University of Science and Technology, Wuhan 430074, China
[b] Key Laboratory of Image Processing and Intelligent Control (Huazhong University of Science and Technology), Ministry of Education, China
[c] Division of Computer Science and Mathematics, University of Stirling, Stirling FK9 4LA, UK

ARTICLE INFO

ABSTRACT

Vehicle scheduling plays a profound role in public transit planning. Traditional approaches for the Vehicle Scheduling Problem (VSP) are based on a set of predetermined trips in a given timetable. Each trip contains a departure point/time and an arrival point/time whilst the trip time (i.e. the time duration of a trip) is fixed. Based on fixed durations, the resulting schedule is hard to comply with in practice due to the variability of traffic and driving conditions. To enhance the robustness of the schedule to be compiled, the VSP based on stochastic trip times instead of fixed ones is studied. The trip times follow the probability distributions obtained from the data captured by Automatic Vehicle Locating (AVL) systems. A network flow model featuring the stochastic trips is devised to better represent this problem, meanwhile the compatibility of any pair of trips is redefined based on trip time distributions instead of fixed values as traditionally done. A novel probabilistic model of the VSP is proposed with the objectives of minimizing the total cost and maximizing the on-time performance. Experiments show that the probabilistic model may lead to more robust schedules without increasing fleet size.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

In public transport, the operation planning process commonly includes the following basic activities, usually performed in sequence: network route design, timetabling (Ibarra-Rojas et al., 2014), vehicle scheduling (Ceder, 2011), crew scheduling and rostering (Shen et al., 2013). Given precompiled timetables, the Vehicle Scheduling Problem (VSP) is concerned with the allocation of a fleet of vehicles to carry out all the trips in the timetables in the most efficient way. An efficient schedule can bring transit operators a considerable saving on property and operating cost.

To clarify the Vehicle Scheduling Problem (VSP), some terminologies need to be introduced. A *trip*, also called a *service trip* in the timetable, is the task unit to be carried out by a vehicle. Each trip contains a start time at its departure point and an end time at its arrival point, and the duration of a trip is called *trip time*. The daily work of a vehicle starts from a *pull-out* from a depot, followed by a sequence of service trips, and ends at a *pull-in* to the depot. The waiting time and empty movement between any two consecutive trips are called an *idle time* and a *deadhead*, respectively. In practice, when the gap between two consecutive service trips is large enough (e.g. more than 3 h), a vehicle is enforced to return to a

---

* Corresponding author. Tel.: +86 27 87540055; fax: +86 27 87543130.
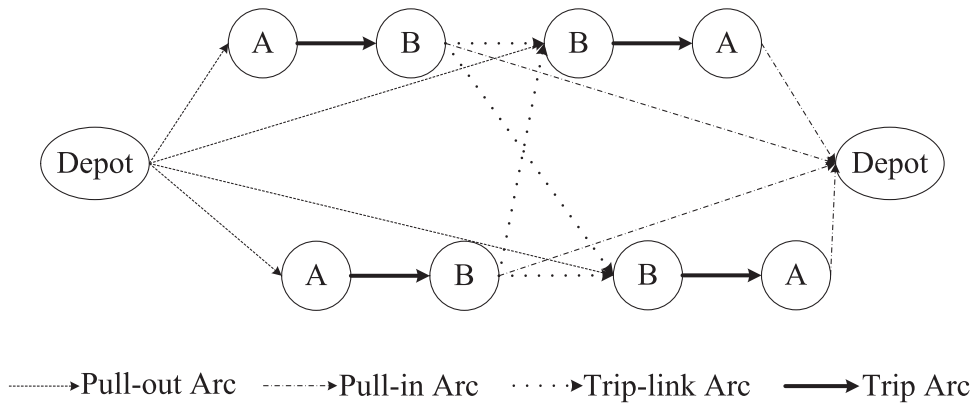 *E-mail address:* yindong@hust.edu.cn, shen.yindong@gmail.com (Y. Shen).

Fig. 1. A network flow model for the traditional VSP.

depot temporarily, which is called a *depot-return*. The objective of VSP is to minimize the total cost including the fixed cost for each used vehicle as well as the variable operation cost including the trip times (duration), idle times and deadheads (Huisman et al., 2004), subject to the following constraints: (1) each trip has to be assigned to exactly one vehicle; (2) each vehicle has to start its first trip following a pull-out from a depot and end its daily work with a pull-in to the same depot; (3) the connection of any two consecutive trips in a vehicle must be time feasible or compatible. Fig. 1 illustrates a network flow model for the traditional VSP, where the letter (A or B) in cycle before a solid arrow denotes the departure time/point of a trip while the letter in cycle after a solid arrow denotes the arrival time/point of a trip.

The VSP has attracted much research interest over the past several decades. Early research focused on the problem with a single depot (SDVSP) (Bodin et al., 1983; Gavish and Shlifer, 1979). Since the 1990s, most research has focused on the problem with multiple depots (MDVSP) (Hadjar et al., 2006; Kliewer et al, 2006), which is proven to be NP hard by Bertossi et al. (1987). Meanwhile, more realistic characteristics are included, such as fuel consumption (Haghani and Banihashemi, 2002), time windows (Desaulniers et al., 1998; Kliewer et al., 2011), multiple vehicle types (Ceder, 2011; Hassold and Ceder, 2014; Kliewer et al., 2006) and integration with crew scheduling (Huisman and Freling, 2005; Shen and Ni, 2006; Shen and Xia, 2009).

The VSP is usually formulated as mathematical optimization models, on which Bunte and Kliewer (2009) gave an overview. In most existing models, each trip is assumed to be fixed (with fixed departure time and arrival time), making the resulting schedule hard to be adhered to in practice due to the dreadful variability of traffic and driving conditions. To reduce the delays, much research has been carried out, which can be roughly classified into the following three groups: (1) to set more accurate scheduled trip times for vehicle scheduling approaches; (2) to adjust schedules or do rescheduling during real-time operations; (3) to enhance the robustness of a vehicle schedule.

The trip time is one of the most essential parameters for the compilation of a vehicle schedule. Its accuracy greatly affects the on-time probability of a vehicle schedule (Shen et al., 2015). This parameter is called *scheduled trip time* in order to be distinguished from the *actual trip time*. There are various ways to set trip times. The early practice is usually conducted by human schedulers, based on their experiences and common sense (Furth and Muller, 2007). In the last decades, automatic data collection systems have been increasingly installed. The large amount of collected data helps towards the design of the more accurate scheduled trip times. In empirical methods, a scheduled trip time is often set as the mean observed trip time (Kittelson and Associates, 2003), the mean trip time minus 2 (Salicrú et al., 2011), the 85th percentile observed running time (Furth, 2006), or the mean value plus the standard deviation (Muller and Furth, 2000). Zhao et al. (2006) proposed a waiting time optimization model for trip time setting, with the objective of minimizing the passenger expected waiting time. Xu and Shen (2012) devised a trip time setting method, aiming to reduce the variation of departure delay. Although these methods help to improve the on-time performance of resulting vehicle schedules, they commonly encounter the following puzzle: a longer trip time tends to increase the cost, while a shorter trip time potentially decreases the punctuality of a schedule.

Apart from setting scheduled trip times in advance, some methods for adjusting schedules or rescheduling during operations have been developed, in which the disturbances affecting trip times are considered. For instances, Huisman et al. (2004) proposed a dynamic vehicle scheduling approach to solve a sequence of rescheduling problems in real-time. A review on more real-time vehicle schedule recovery methods can be found in Visentini et al. (2014).

To reduce real-time recovery activities, some efforts are put into enhancing the robustness or delay-tolerance of pre-compiled schedules. Kramkowski et al. (2009) proposed a heuristic approach to redistribute the links (i.e. break times) of a given cost-optimal vehicle schedule, in which a first-in-first-out decomposition strategy is applied to even the links between consecutive trips. Their experimental results show that the delay-tolerance of given vehicle schedules can be enhanced. Later, Amberg et al. (2011) proposed two decomposition strategies of local decomposition and global decomposition for redistributing the links, in which the global decomposition utilizes the delay scenarios published in Huisman et al. (2004). The proposed method is applied to vehicle scheduling and integrated scheduling of vehicle and crew, and the resulting

schedules show higher tolerance to delay without increasing the costs of given pre-compiled schedules. However, the degree of improvement on the delay-tolerance (or robustness) of a schedule is considerably limited by given original schedules. To compile a robust vehicle schedule from scratch, Naumann et al. (2011) proposed a stochastic programming approach, aiming to minimize the expected cost consisting of planned costs and disruption costs caused by delays.

Although the existing robust VSP approaches can increase the delay-tolerance of resulting vehicle schedules to a certain extent, they are all based on a set of fixed trips (with fixed departure times and durations) which implied that the, resulting schedules are still hard to adhere to in practice due to the large variability of traffic and driving conditions. Moreover, in their VSP model based on fixed trips, the cost of a trip-to-trip link is defined by a fixed link cost (consists of idle time and deadhead) plus a fixed penalty of delay. Therefore, for every linked two trips assigned to one vehicle as two consecutive tasks, the arrival time of the anterior trip will not affect the departure time of the posterior trip, nor its arrival time.

To enhance the robustness of the vehicle schedule generated, we study the static VSP in a different way by considering the trips with scheduled departure times and stochastic durations. The stochastic trip times follow the probability distributions obtained from the data captured by Automatic Vehicle Locating (AVL) systems.

The remainder of the paper is structured as follows. Section 2 proposes a probabilistic model for VSP, in which each trip time is defined as a random variable. Furthermore, the model is enhanced by considering *delay propagation* from one trip to its next consecutive trip. Section 3 develops a solution method, and Section 4 displays experimental results. Some concluding remarks and possible future work are given in Section 5.

2. Robust vehicle scheduling based on stochastic trip times

With stochastic trip times, the VSP cannot be represented clearly by Fig. 1, therefore, we proposed a new network flow representation in this section before building a probabilistic model for the VSP with stochastic trip times.

2.1. Network flow representation for VSP with stochastic trip times

A new network flow model is built for VSP with stochastic trip times, which can be denoted as a directed graph $G = (N, A)$. The set of nodes N contains two types of nodes: *trip nodes* and *depot nodes*. The difference from the traditional VSP model is that a trip node corresponds to a trip with a scheduled departure time and a stochastic trip time characterized by its trip time probability distribution. The set of arcs A contains four types of arcs: *pull-out arcs, pull-in arcs, depot-return arcs* and *trip-link arcs*. A *pull-out arc* connects a depot to a trip, a *pull-in arc* connects a trip to a depot, a *depot-return arc* is a long arc (longer than a given time, e.g. 3 h) connecting two trip nodes with a temporary return to a depot in between, and *trip-link arcs* are the other arcs connecting two trip nodes. Notice that a *depot-return arc* is actually a special case of a *trip-link arc*, and we distinguish them due to the difference of their costing methods applied commonly in practice. The cost of a *trip-link arc* consists of an idle time and a deadhead time (if exist) between two connected trips, while the cost of a *depot-return arc* consists of a deadhead time to the depot and a deadhead time back to work from the depot. The time that vehicle spends temporarily at depot is not costed as an idle time.

Fig. 2 illustrates the new network flow model, where A → [B] or B → [A] represents a trip node, the letter in brackets denotes an arrival point with a stochastic arrival time while the letter without brackets denotes a departure point with a scheduled departure time.
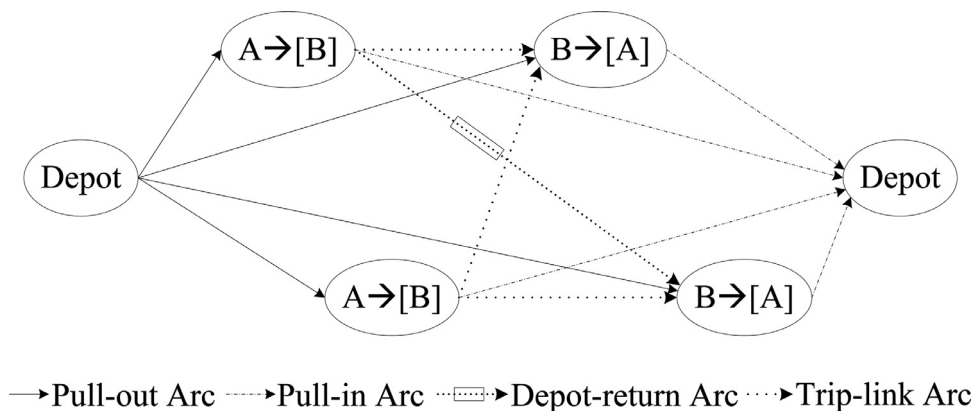

Fig. 2. A new network flow model for the VSP with stochastic trip times.
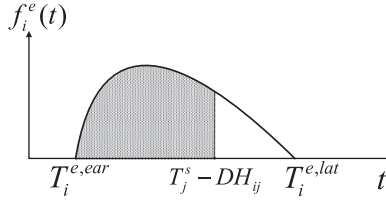
Fig. 3. Compatibility probability of any two trips i and j.

2.2. Compatibility probability

The generation of arcs is essential for implementing the network flow model for the VSP. An arc between any pair of trips can be generated if the two trips are compatible. In the traditional VSP model, the compatibility of any pair of trips can be simply defined as time feasibility, which is deterministic. Let T_i^s and T_i^e denote the scheduled departure time and arrival time of trip i, respectively. Bertossi et al. (1987) defined the compatibility $i\Theta j$ of any two trips i and j with fixed trip times as

$$T_i^e + DH_{ij} \le T_j^s, \tag{1}$$

where DH_{ij} denotes the deadhead time from the arrival point of trip i to the departure point of trip j. If two trips i and j are compatible, the arc (i, j) exists and its cost C_{ij} can be set as the sum of the deadhead and idle time in between.

In our research, we consider the given trips with stochastic trip times. Consequently, the compatibility of any pair of trips becomes uncertain, and the compatibility probability is defined to determine the existence of the arcs between trips. Let stochastic variables t_i^s, t_i^e and t_i be the departure time, arrival time and duration of trip i, respectively. Suppose the duration of trip i (i.e. the stochastic trip time t_i) follows a known Probability Density Function (PDF) $f_i(t)$ and $t_i \in [T_i^{\min}, T_i^{\max}]$, where $T_i^{\min}$ and $T_i^{\max}$ denote the minimum and maximum trip times of trip i respectively. A compatibility probability of two trips i and j can be defined as illustrated in Fig. 3, where the curve denotes the arrival time distribution $f_i^e(t) = f_i(t - T_i^s)$ of trip i. The stochastic arrival time $t_i^e \in [T_i^{e,ear}, T_i^{e,lat}]$, where $T_i^{e,ear} = T_i^s + T_i^{\min}$ and $T_i^{e,lat} = T_i^s + T_i^{\max}$ denote the earliest and latest arrival times of trip i respectively. Notice that the deadhead time DH_{ij} is stochastic in actual operation, which can be defined as a stochastic variable and treated similarly as the stochastic trip time. However, to concentrate on handling stochastic trip times, the DH_{ij} is assumed to be a deterministic value in this paper.

Given two trips i and j with stochastic trip times, we define the compatibility $i\Theta j$ as $t_i^e \le T_j^s - DH_{ij}$. Therefore, the compatibility probability $P\{i\Theta j\}$ can be defined as the probability of the random event that i and j are compatible, i.e. $P\{i\Theta j\} = P\{t_i^e \le T_j^s - DH_{ij}\}$, which corresponds to the grey area shown in Fig. 3 and can be expressed as

$$P\{i\Theta j\} = \int_0^{T_j^s - DH_{ij}} f_i^e(t)dt = \int_0^{T_j^s - T_i^s - DH_{ij}} f_i(t)dt, \tag{2}$$

where $P\{i\Theta j\} = 0$ if $T_j^s - DH_{ij} < T_i^{e,ear}$ and $P\{i\Theta j\} = 1$ if $T_j^s - DH_{ij} > T_i^{e,lat}$.

Between the trips i and j, an arc (i, j) is defined if $P\{i\Theta j\} > 0$; no arc exists, otherwise. Each arc (i, j) is also associated with an incompatibility probability $P\{i\overline{\Theta} j\} = 1 - P\{i\Theta j\}$, where $i\overline{\Theta} j$ denotes the incompatibility of trips i and j.

As already mentioned, a *depot-return* is enforced in practice, when the gap between two consecutive service trips is large enough. Let T^{dr} be the minimal time required for a temporary depot return at a depot, DH_{id} be the deadhead time from the arrival point of trip i to depot d, and DH_{dj} be the deadhead time from depot d to the departure point of trip j. An arc (i, j) is defined as a *depot-return arc* if $T_j^s - T_i^{e,lat} - (DH_{id} + DH_{dj}) \ge T^{dr}$, otherwise it is defined as a *trip-link arc*. For any *depot-return arc* (i, j), its compatibility probability is defined as $P\{i\Theta j\} = 1$.

2.3. A probabilistic model for VSP with stochastic trip times

To establish our probabilistic model for VSP with stochastic trip times, a cost function for any arc (i,j) is first defined as follows:

$$C_{ij} = \begin{cases} DH_{ij} \text{ if } (i, j) \text{ is a pull} - \text{out arc, } i \in D, j \in T \\ DH_{ij} \text{ if}(i, j)\text{is a pull} - \text{in arc, } i \in T \text{ and } j \in D \\ DH_{id} + DH_{dj} \text{ if}(i, j)\text{is a depot} - \text{return arc, } i, j \in T, d \in D \\ DH_{ij} + E(ID_{ij}) \text{ if}(i, j)\text{is a trip} - \text{link arc, } i, j \in T \end{cases} \tag{3}$$

where T denotes the set of trips, D denotes the set of depots and DH_{ij} denotes the deadhead between trips i and j. For the cost of a *depot-return* arc, the time spent temporarily at depot is not included as usual.

The cost of a *trip-link arc* (i, j) consists of a *deadhead time* DH_{ij} which is a deterministic value and an *idle time* ID_{ij} which is a stochastic variable. If $i\Theta j$, then $ID_{ij} = T_j^s - t_i^e - DH_{ij}$; otherwise, set ID_{ij} to be 0. As ID_{ij} is a random variable, it cannot be directly minimized. We therefore use its expected value $E(ID_{ij})$ as the cost of idle time formulated as

$$E(ID_{ij}) = \int_0^{T_j^s - DH_{ij}} (T_j^s - t_i^e - DH_{ij}) f_i^e(t) dt = \int_0^{T_j^s - T_i^s - DH_{ij}} (T_j^s - T_i^s - t - DH_{ij}) f_i(t) dt. \tag{4}$$

As already defined, an arc exists if $P\{i\Theta j\} > 0$ although it may be time infeasible. Hence, we have to penalize the infeasibility of the arcs. Let IF_{ij} be the infeasible time of an arc (i, j). If $i\bar{\Theta} j$, then $IF_{ij} = t_i^e - T_j^s + DH_{ij}$; otherwise $IF_{ij} = 0$. Clearly, IF_{ij} is also a random variable which cannot be minimized directly, and thus we can minimize its expected value $E(IF_{ij})$. However, to even the infeasible time amongst the arcs, we penalize the stochastic infeasible time by using $E(IF_{ij}^2)$ instead of $E(IF_{ij})$, as suggested in Huisman et al. (2004) and Naumann et al. (2011). Therefore, the penalty of the infeasible time of arc (i,j) $P_{ij} = E(IF_{ij}^2)$ is defined as

$$E(IF_{ij}^2) = \int_{T_j^s - DH_{ij}}^{+\infty} (t_i^e - T_j^s + DH_{ij})^2 f_i^e(t) dt = \int_{T_j^s - T_i^s - DH_{ij}}^{+\infty} (T_i^s + t - T_j^s + DH_{ij})^2 f_i(t) dt \tag{5}$$

Let P denote the set of *pull-out* arcs, Q denote the set of *pull-in* arcs and R denote the union of the set of *trip-link* arcs and the set of *depot-return* arcs. Given a set of arcs $A = P \cup Q \cup R$, the VSP with stochastic trip times can be modelled as

$$\min \sum_{(d,j)\in P} C_{veh} x_{dj} + \sum_{(i,j)\in A} C_{ij} x_{ij} + \alpha \cdot \sum_{(i,j)\in R} P_{ij} x_{ij} \tag{6}$$

$$\text{s.t.} \sum_{i:(i,j)\in A} x_{ij} - \sum_{i:(j,i)\in A} x_{ji} = 0, \quad \forall j \in T \tag{7}$$

$$\sum_{j:(d,j)\in P} x_{dj} - \sum_{i:(i,d)\in Q} x_{id} = 0, \quad \forall d \in D \tag{8}$$

$$\sum_{i:(i,j)\in A} x_{ij} = 1, \quad \forall j \in T \tag{9}$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A \tag{10}$$

where decision variable x_{ij} equals 1 if arc (i, j) is selected; 0 otherwise. Formulae (7)–(10) are constraints the same as those in traditional network flow models of VSP with fixed trip times. Formula (7) and Formula (8) are the flow-conservation constraints. Formula (9) is the cover constraints requiring that each trip must be covered exactly by one vehicle, and Formula (10) is the 0–1 constraint for decision variables. Formula (6) is the objective function, where C_{veh} is a large constant to penalize the utilization of an additional vehicle, and α is a non-negative (usually larger than 1) weight to adjust the penalty of infeasible time. The objective function may serve as a trade-off between our two conflict objectives: one is to minimize the operating cost while another is to maximize the on-time performance.

From the vehicle scheduling point of view, the on-time performance referring to the punctual departure of trips can be measured by departure delay. A vehicle schedule that may cause less departure delays is defined as a *robust schedule* in this study.

In the traditional network flow model of VSP with fixed trip times, minimizing the cost is the only objective since infeasible arcs are not allowed at all. However, in our proposed model, with the aid of given trip time probability, an arc is defined if $P\{i\Theta j\} > 0$ although it may be time infeasible. This provides more opportunities for selecting long arcs without increasing the cost. With the same cost (mainly determined by the fleet size) and the same trip time distributions, the longer the arcs are, the more robust a schedule is. Therefore, the resulting schedule based on the probabilistic model may be more robust.

2.4. An enhanced probabilistic model for VSP featuring delay propagation

Given any pair of trips (i, j) that are successively worked by one vehicle, suppose that trip i departs on time at its scheduled departure time T_i^s, and the PDFs for trip time and arrival time of trip i are $f_i(t)$ and $f_i^e(t) = f_i(t - T_i^s)$ respectively. If trip i arrives on or before $T_j^s - DH_{ij}$ (i.e. $i\Theta j$), trip j can depart on time at its scheduled departure time T_j^s; otherwise, the departure time of trip j has to be postponed or delayed. Such a delay may propagate from one trip to its next consecutive trip, which is called *delay propagation*. During the execution of a vehicle schedule, the delay on a trip can propagate perpetually until it is absorbed by an idle time (i.e. the next trip can depart on time) or the daily work of the vehicle ends.

It should be noticed that the delays amongst different vehicles may be correlated in real-time operation. As Xuan et al. (2011) indicated bus systems are naturally unstable in the sense that a small disturbance such as a traffic disruption can start a vicious cycle that causes bus bunching. However, in this study, the delay propagation to multiple trips among different vehicles is not considered as otherwise the problem will become extremely complex.

Studying the delay propagation amongst the trips on the same vehicle, the departure time of trip j becomes a stochastic variable affected by the arrival probability of trip i, instead of a fixed scheduled departure time. The PDF for departure time of trip j can be expressed as a piecewise function of $f_i^e(t)$ formulated as

$$f_j^s(t) = \begin{cases} 0 & \text{if } t < T_j^s \\ \int_0^{T_j^s - DH_{ij}} f_i^e(t)dt & \text{if } t = T_j^s \\ f_i^e(t - DH_{ij}) & \text{if } t > T_j^s \end{cases} \tag{11}$$

Meanwhile, the delay on departure time of trip j may further postpone its arrival time. Therefore, it is no longer adequate to calculate the PDF for arrival time of trip j as $f_j^e(t) = f_j(t - T_j^s)$. A new expression featuring delay propagation can be as Formula (12), which is a function of its own PDFs for the departure time and the trip time.

$$f_j^e(t) = \int_0^t f_j^s(\tau) f_j(t - \tau | t_j^s = \tau) d\tau \tag{12}$$

Using Formula (11) to substitute $f_j^s(t)$, the PDF for the arrival time of trip j is

$$f_j^e(t) = \left[\int_0^{T_j^s - DH_{ij}} f_i^e(t)dt \right] f_j(t - T_j^s | i\Theta j) + \int_{T_j^s}^t f_i^e(\tau - DH_{ij}) f_j(t - \tau | i\overline{\Theta} j) d\tau. \tag{13}$$

From Formula (13), we can see that the arrival time probability of any trip is the function of its own trip time probability and the arrival time probability of its previous trip. Obviously, this is a recursive function.

Therefore, the cost function for any arc (i,j) can remain the same as Formula (3), where the expected idle time $E(ID_{ij})$, representing the cost of the idle time, is expressed as

$$E(ID_{ij}) = \int_0^{T_j^s - DH_{ij}} (T_j^s - t_i^e - DH_{ij}) f_i^e(t)dt, \tag{14}$$

where $f_i^e(t)$ is the PDF for the arrival time of trip i as presented in Formula (13).

To express more straightforwardly the delay propagation associated to any arc (i, j), a *departure delay* (DD_{ij}) is employed instead of the infeasible time (IF_{ij}) used in the previous section. If $i\overline{\Theta} j$, then $DD_{ij} = t_i^e - T_j^s + DH_{ij}$; otherwise $DD_{ij} = 0$, where t_i^e is the stochastic time decided by Formula (13). Similarly, to even the *departure delay* amongst the arcs, we penalize the stochastic *departure delay* of an arc (i, j) by using the *expectation of squared departure delay*, which is defined as $P_{ij} = E(DD_{ij}^2)$ and expressed as

$$E(DD_{ij}^2) = \int_{T_j^s - DH_{ij}}^{+\infty} (t_i^e - T_j^s + DH_{ij})^2 f_i^e(t)dt. \tag{15}$$

Consequently, the enhanced model is built, which can be expressed the same as in Formulae (6)–(10), but has considerable features. In the previous probabilistic model, the cost or the penalty of any trip-link arc (i,j) contains only one stochastic variable, i.e. the trip time probability of trip i. Therefore, the cost C_{ij} and the penalty P_{ij} can be determined when the trip time probability of trip i is determined. However, in this enhanced model featuring delay propagation, the C_{ij} and P_{ij} are the functions of the arc (i,j) and its previous arcs. Unfortunately, the previous arcs cannot be determined before a schedule is produced. Hence, this model cannot be solved directly by the existing vehicle scheduling approaches, which are all based on the deterministic model. To solve the model, a heuristic approach is developed in the following section.

3. Solution method

This section proposes a hybrid approach for vehicle scheduling, which integrates Integer Linear Programming (ILP) approaches and heuristic methods. The basic idea is as follows: an initial schedule is first compiled using a matching based heuristic, and then is refined by an iterative greedy local search.

3.1. Compilation of an initial schedule

A matching based heuristic method, proposed by Ball et al. (1983) for the integrated vehicle and crew scheduling problem, is adapted to produce an initial schedule for the VSP. The method consists of two stages: tier partitioning and trip matching.

The tier partitioning stage is to partition all the trips, which are sorted by departure times in advance, into different tiers. Each trip is considered in turn until it is classified into a certain tier. The first trip belongs to tier 1. For any unclassified trip j, a backward search is applied, which starts from the currently largest existing tier L to tier 1 by the following steps:

Step 1: Let $l = L$;
Step 2: If a trip i in tier l satisfying $P\{i\Theta j\} > 0$ can be found, trip j is classified into tier $l+1$ and stop;
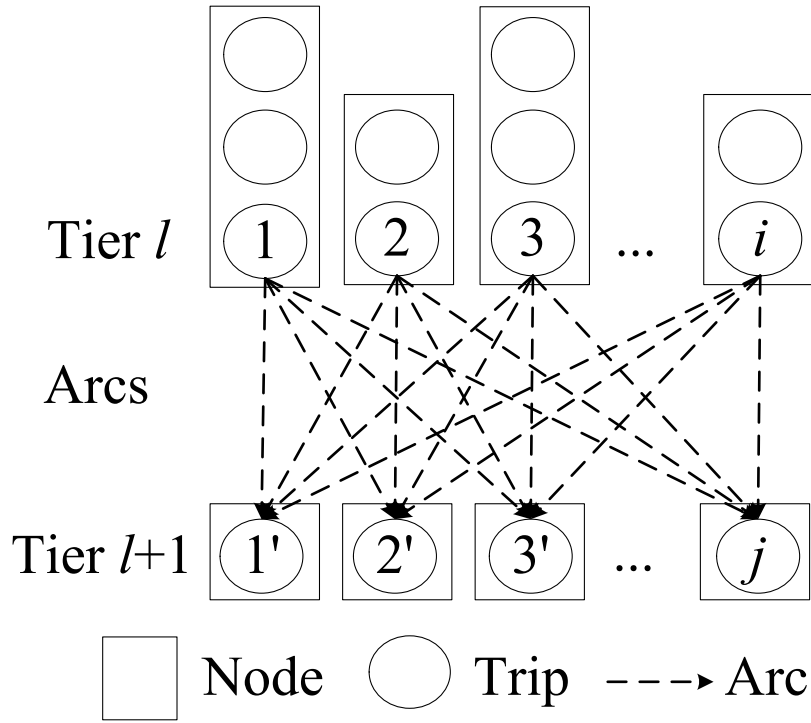
Fig. 4. Illustration of a matching problem between two adjacent tiers.

Step 3: If $l = 1$, trip j is classified into tier 1, which means that it can only be operated as the first trip of a vehicle; otherwise let $l = l – 1$ and go to Step 2.

The trip matching stage is to form and solve a series of matching problems for any tier (from the first to the last second tier) with its successive tier in sequence for linking trips into an initial schedule. Given two adjacent tiers l and $l + 1$, a matching problem is built by forming a bipartite graph as illustrated in Fig. 4, where each tier contains a set of nodes. A node in Tier l is a trip-chain consisting of the trips in Tier l and its previous tiers; a node in Tier $l + 1$ corresponds to a trip. The arcs represent all the potential links between the nodes in the two tiers.

The matching problem is concerned with the selection of a set of arcs with the minimum cost, subject to that no selected arcs can share the same node. The solution is a set of new trip chains, which constitutes the new nodes in Tier $l + 1$. An initial schedule is produced when the matching problem regarding the last two tiers is solved.

3.2. Solution refinement

After the initial solution is constructed, an iterative greedy local search method is proposed to refine the schedule. The basic idea is to break part of the arcs in the current schedule and then to re-link the nodes by solving the sub-problem containing a small subset of nodes using an exact ILP approach. Such a procedure will repeat iteratively until a termination criterion is met. Two methods for breaking and re-linking are developed and applied in sequence.

The first method contains the following steps. Suppose the current schedule contains n vehicles denoted by $V = (v_1, v_2, \ldots, v_n)$. Each vehicle v_i has an associated evaluation value c_i, which is defined as the sum of the evaluation values (consisting of costs and penalties) of all the arcs on the vehicle.

Step 1: Let $i = 1$ and $j = i + 1$;
Step 2: Select a pair of vehicles v_i and v_j;
Step 3: Form a sub-problem G_k using all the nodes and their corresponding arcs in v_i and v_j;
Step 4: Solve G_k and get a new partial schedule S' with evaluation value c', which may contain one or two new vehicles v_i' and v_j';
Step 5: If $c' < c_i + c_j$, then update the vehicle list V by moving v_i' and v_j' to the tail of V and removing v_i and v_j from V, and then let i remain the same (but v_i will represents a different vehicle), let $j = i+1$, go to Step 2; otherwise, go to Step 6;
Step 6: If $j < n$, let $j = j+1$ and go to Step 2; otherwise, go to Step 7;
Step 7: If $i < n -1$, let $j = i+2$ and $i = i+1$, go to Step 2; otherwise, stop.
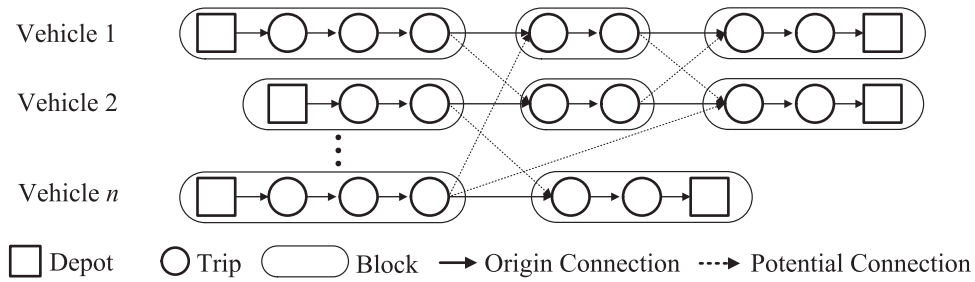
Fig. 5. Illustration of the refinement method with destruction and reconstruction.

The second method contains three operators: *arc selection, destruction* and *reconstruction*. A set of arcs are first selected according to the following criteria: the first is to select all the depot-return arcs, which are undesired by operators, the second is to select the other undesired arcs with the largest costs or penalties, the third is to select randomly a certain number of arcs. The total amount of selected arcs is restricted to a certain number (say 50 in this case to fit with an ILP solver's ability). Then, by breaking the selected arcs, the current schedule is destroyed into a set of *blocks* (a block contains several consecutive trips). Next, a new problem is formed, in which each block is regarded as a node. The reconstruction step re-links the blocks by solving a network flow problem. The method is demonstrated in Fig. 5.

After the reconstruction at each iteration, a new schedule S' will be generated from the previous schedule S. Let $F(S)$ and $F(S')$ denote the objective values of S and S', respectively. S' is accepted if $F(S') < F(S)$; otherwise S' is accepted at probability $exp\{[(F(S) - F(S'))]/T_n\}$, where $T_n = \alpha * T_{n-1}$ represents the temperature at the nth iteration, T_0 is the initial temperature and α is the cooling rate. This method terminates if schedule cannot be improved for a certain number of iterations or a maximum number of iterations has been reached.

4. Experiments and results

A series of experiments have been carried out based on a case study on Route 4 of Haikou Bus (HKB4). HKB4 is a key route of Haikou city in China, which operates over 450 trips and carries around 46,000 passengers per day. 28 buses are currently deployed on the route and all of them are equipped with AVL devices. The scheduled headway is about 4 to 5 min at most time throughout the day, with a maximum of 6 min in the afternoon off-peak period.

Experiments on the traditional deterministic VSP model (*DVSP* for short) are first presented in Section 4.1. Experiments on the probability VSP model without considering delay propagation (*PVSP* for short) are then presented in Section 4.2. At the end, experiments on the probability VSP model considering delay propagation (*PVSP-DP* for short) are presented in Section 4.3.

4.1. Experiments on the deterministic model with fixed trip times (DVSP)

This section aims to generate benchmark schedules based on the DVSP model. CPLEX, a widely used commercial mathematical programming optimizer, is employed to solve the model.

Scheduled trip times are essential for the DVSP model. To automatically set a series of scheduled trip times, the trip time distributions need to be obtained in advance by the following steps: a sample set of the trip times is firstly extracted from historical AVL data, then Homogeneous Running Time (HRT) periods, each of which holds the trip times with an identical distribution, are calculated by the approach proposed in Xu and Shen (2012), and finally the trip time probability distribution within each HRT period is calculated. In the case of HKB4, the AVL data was recorded during May to August 2010, and a sample set of 11585 trips were formed. Table 1 shows the HRT periods and the corresponding trip times, where AVG and STD denote the average trip time and the standard deviation.

Next, the scheduled trip time for each HRT period can be generated, which is commonly decided by rules of thumb (Furth, 2006).The following two rules-of-thumb (Muller and Furth, 2000) are often used by schedulers in practice: the first one uses the 85th percentile of the entire trip time (i.e. 85% of scheduled trips are expected to be able to completed on time), and the second one uses the average value (AVG) plus standard deviation (STD) of trip times. Two groups of the trip times are produced as shown in Table 2.

To produce more benchmark schedules, we adjust the two groups of trip times in Table 2 with +/– 1 min (1 min is commonly the minimum unit in a vehicle schedule). Therefore, six groups of scheduled trip times are produced, which correspond to six VSP instances (T1 to T6) as shown in Table 3, where all the times are in minutes, T1 to T3 are based on the *rule-of-thumb 1*, and T4 to T6 are based on the *rule-of-thumb 2*. Table 3 also presents the computational results generated by CPLEX 12.4, where the *service trip time* is the sum of all scheduled trip times in the corresponding schedule. Note that during the scheduling process, a 2-min recovery time is set at the end of each trip to improve the on-time performance of the resulting schedule.

Table 1
HRT periods of HKB4 obtained based on the AVL data.

	Outbound					Inbound			
			Trip time						Trip time	
HRT #	Period start	Period end	AVG	STD	HRT #	Period start	Period end	AVG	STD
1	05:45	07:15	43	2.2	1	05:45	07:30	43	4.2
2	07:15	07:30	48	1.9	2	07:30	08:15	48	3.7
3	07:30	11:45	51	3.0	3	08:15	11:15	45	4.0
4	11:45	14:15	48	2.3	4	11:15	11:30	48	4.9
5	14:15	16:45	54	3.0	5	11:30	14:00	43	4.8
6	16:45	17:00	59	4.4	6	14:00	17:15	48	4.5
7	17:00	18:30	63	4.1	7	17:15	18:00	53	4.2
8	18:30	18:45	59	3.9	8	18:00	18:30	49	5.0
9	18:45	20:15	53	3.0	9	18:30	21:45	45	4.7
10	20:15	21:00	57	3.6	10	21:45	22:15	42	3.3
11	21:00	22:00	61	4.4	11	22:15	23:30	36	2.8
12	22:00	23:30	55	4.2					

Table 2
Fixed trip times produced by rules-of-thumb.

Rule-of-thumb 1 (85%)				Rule-of-thumb 2 (AVG+STD)			
HRT # (Outbound)	Trip time (min)	HRT # (Inbound)	Trip time (min)	HRT # (Outbound)	Trip time (min)	HRT # (Inbound)	Trip time (min)
1	46	1	46	1	45	1	47
2	50	2	51	2	50	2	52
3	54	3	47	3	54	3	49
4	51	4	48	4	50	4	53
5	57	5	45	5	57	5	48
6	62	6	49	6	63	6	52
7	67	7	56	7	67	7	57
8	62	8	53	8	63	8	54
9	56	9	47	9	56	9	50
10	59	10	44	10	61	10	45
11	64	11	39	11	65	11	39
12	59			12	59		

Table 3
Experimental results of DVSP – produced by CPLEX.

Problem	Rule-of-thumb	Trip time adjustment	Fleet size	Service trip time	E(ID)	Dead-head	Cost	Penalty	Objective value
T1	Rule-of-thumb 1	−1	27	22816	2313	1980	4293	9577	45659
T2		0	28	23270	2698	2040	4738	6713	42807
T3		1	29	23724	3372	2100	5472	3497	39718
T4	Rule-of-thumb 2	−1	28	23290	2762	2100	4862	5637	41317
T5[a]		0	<u>28</u>	<u>23744</u>	<u>3315</u>	<u>2040</u>	<u>5355</u>	<u>3627</u>	<u>38796</u>
T6		1	29	24198	3611	2250	5861	2769	39015

[a] The *best-known schedule.*

From Table 3, we can see that the service trip time of each schedule is different due to the adjustments of +/− 1 min to the scheduled trip times. To compare the schedules produced by all the experiments in this paper with a consistent measurement, the objective value of Formula (6) is employed, where $\alpha = 1.5$ is a constant designated based on a series of experiments, and the enhanced probabilistic model for VSP featuring delay propagation is employed since delay propagations exist in real-time operation. Moreover, to give some insight into the objective value, each of its elements is also displayed, including the feet size (assuming each vehicle costs 1000 min), the cost and the penalty. The cost is the sum of the cost of each arc defined by Formula (3), where the $E(ID)$ is calculated using Formula (14) and the deadhead is the sum of all kinds of deadhead. The penalty (i.e. $E(DD^2)$) is calculated using Formula (15), which is served as the measurement of on-time performance.

We can see from Table 3 that the six results of T1 to T6 show trade-offs between the cost and penalty. Let us discuss which one should be regarded as the best benchmark schedule from the view of human schedulers. In vehicle scheduling, minimizing the fleet size with reasonable on-time performance (measured by the penalty) is the overriding objective. Although T1 has the smallest fleet size, it cannot be regarded as the best or suitable schedule since its on-time performance is too low to be acceptable in practice. In fact, the currently deployed fleet size is 28 vehicles in this case. Meanwhile, T3

Table 4
Experimental results of PVSP (produced by CPLEX).

Index	α	Based on the PVSP model				RPD over the best-known schedule		
		Fleet size	Cost	Penalty	Objective value	Cost	Penalty	Objective value
R1	0.2	27	5394	6098	41541	0.73%	68.11%	7.08%
R2	0.3	28	5102	3708	38664	−4.72%	2.22%	−0.34%
R3	0.4	28	5107	3693	38646	−4.63%	1.80%	−0.39%
R4	0.5	28	5215	3461	38406	−2.61%	−4.59%	−1.01%
R5	0.6	28	5271	3358	38308	−1.57%	−7.43%	−1.26%
R6	0.7	28	5316	3288	38248	−0.73%	−9.35%	−1.41%
R7	0.8	28	5409	3159	38147	1.01%	−12.91%	−1.67%
R8	0.9	28	5475	3082	38098	2.24%	−15.03%	−1.80%
R9	1	28	5495	3060	38085	2.61%	−15.64%	−1.83%
R10	1.1	28	5505	3050	38080	2.80%	−15.92%	−1.85%
R11	1.2	28	5587	2978	38054	4.33%	−17.90%	−1.91%
R12	1.3	28	5623	2949	38046	5.00%	−18.71%	−1.93%
R13	1.4	28	5684	2903	38038	6.14%	−19.98%	−1.95%
R14	1.5	28	5772	2842	38035	7.79%	−21.65%	−1.96%
R15	1.6	29	5929	2110	38094	10.72%	−41.83%	−1.81%
Avg.		28	5459	3316	38433	2.03%	−8.59%	−0.94%

and T6 are also out of the consideration due to one more vehicle is needed. With the same suitable fleet size, minimizing penalty has usually priority over minimizing cost. Therefore, in consideration of a trade-off between the cost and on-time performance, T5's schedule can be regarded as the *best-known schedule* of this case since it has the highest on-time performance with an expected fleet size. This result is also consistent with that of comparing the objective value, i.e. T5's schedule has the lowest objective value.

4.2. Experiments on the probabilistic model without considering delay propagation (PVSP)

To verify the effectiveness of the PVSP model, a series of experiments are carried out on the HKB4 problem. The schedules are also produced by CPLEX and to be compared with the best-known schedule obtained based on the DVSP model.

The PVSP model contains a non-negative empirical parameter α as shown in Formula (6). The choice of α value is a trade-off between the cost and on-time performance of the resulting schedule. If α is set to 0, the PVSP model is equivalent to the DVSP model with the scheduled trip times setting as the minimum trip times. In this case, a feasible schedule with minimum cost can be obtained, but the on-time performance of the schedule is usually very low. Along with the increment of α value, the on-time performance of the resulting schedule will be enhanced while the cost is compromised. Since minimizing the number of vehicles is the overriding objective, the best α value should be able to keep the highest on-time performance using a reasonable small fleet size. Such a value is problem-specific but can be obtained from experiments. In this HKB4 instance, 15 values of α within the range of 0.2 to 1.6 are investigated, which are indexed from R1 to R15. The compiled schedules are listed in Table 4, where the Relative Percentage Deviations (RPDs) over the best-known schedule (T5's schedule in Table 3) are provided, and all the times are in minutes.

It can be seen from Table 4 that with the increment of α values, the costs are increased while the penalties are decreased, except for R1. With the smallest $\alpha = 0.2$, the R1's schedule has a smallest fleet size but the highest penalty, which means its on-time performance is the worst. In this case, the R1's schedule cannot be executed properly in practice, and thus it will be removed. As for R15, the schedule has the highest on-time performance due to using the largest $\alpha = 1.6$, but it should also be removed since one more vehicle is needed than expected. Therefore, in this case, α should be selected from the range between 0.3 and 1.5, where all the schedules from R2 to R14 have the same fleet size with different operating costs. When α reaches the largest value (i.e. $\alpha = 1.5$), R14 corresponds to the schedule with the highest on-time performance, hence it is regarded as the best schedule in Table 4 and 1.5 is the best α value selected for this problem instance. Therefore, we can conclude that given a suitable α value, the PVSP model may surpass considerably the DVSP model in terms of on-time performance with the same fleet size.

Furthermore, Fig. 6 gives an insight into the schedules in consideration of the two conflict items: cost and penalty, where only the schedules with expected number of vehicles (i.e. 28 vehicles) are compared. Fig. 6 shows the Pareto fronts for the DVSP and PVSP models, where the non-dominated schedules for both models are cycled.

From Fig. 6, we can see that the *best-known schedule* T5 of the DVSP is dominated by the results of the PVSP, while the other two schedules are non-dominated but have considerably larger penalties. All of the PVSP's results are non-dominated solutions, and have considerably better on-time performance (measured by the penalty) than the DVSP's results.

4.3. Experiments on the enhanced probabilistic model with delay propagation (PVSP-DP)

Since the existing approaches are all based on the deterministic model, which cannot be used to solve the PVSP-DP model directly, we proposes a new heuristic approach and implement it using C# on a 2.0 GHz PC with 2 GB RAM under
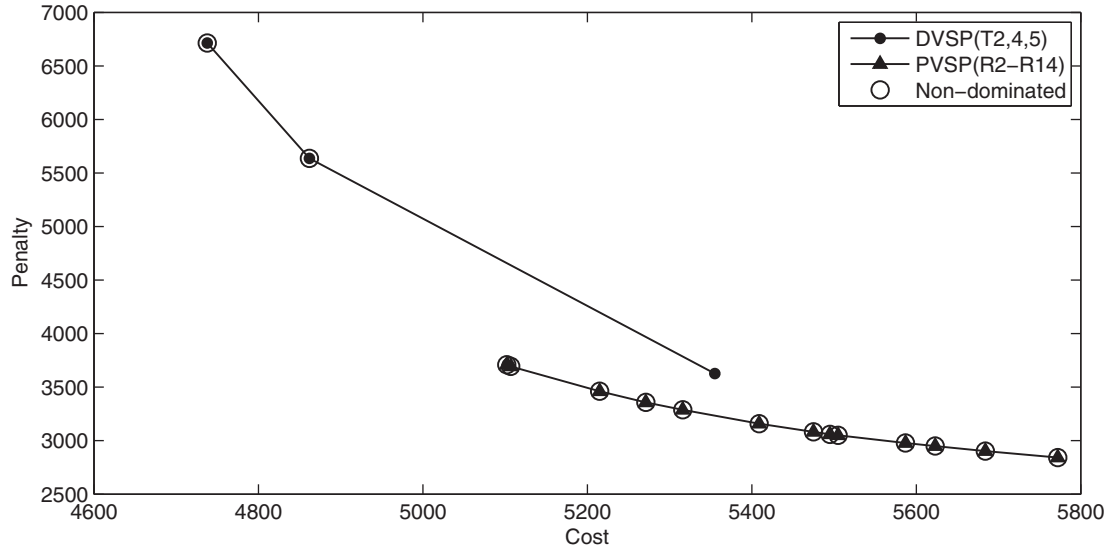
Fig. 6. Results produced by CPLEX based on the DVSP and PVSP models.

Table 5
Experimental results of DVSP (produced by CPLEX and the heuristic approach).

Test problem	Fleet size		Cost			Elapse time (s)	
	CPELX	Heuristic	CPLEX	Heuristic	RPD	CPLEX	Heuristic
T1	27	27	4293	4500	4.82%	335	1581
T2	28	28	4738	5042	6.42%	333	1608
T3	29	29	5472	6130	12.02%	340	1592
T4	28	28	4862	5148	5.88%	331	1588
T5	28	28	5355	5669	5.86%	336	1578
T6	29	29	5861	6712	14.52%	327	1582
Avg.	27	27	5096	5533	8.25%	334	1588

Windows XP, on which the CPLEX 12.4 is employed to solve the embedded small ILP problems. The proposed heuristic approach is first tested and then applied to the two proposed probabilistic models of PVSP and PVSP-DP.

4.3.1. The proposed heuristic approach vs. CPLEX

Based on the DVSP model, this section is to verify the effectiveness of the proposed heuristic approach by comparing its schedules with those produced by CPLEX for problems T1–T6 listed in Table 3. The comparing results are shown in Table 5.

It can be seen from Table 5 that the fleet size of the schedules generated by the heuristic approach are the same as the schedules generated by CPLEX for all the six problems, while the average RPD in terms of cost is 8.25%. Since minimizing the fleet size is an overriding objective for the DVSP, the results produced by the proposed heuristic approach are good enough, and thus this approach is qualified to be employed to evaluate the probabilistic models.

We can also see from Table 5 that the average elapsed time for the heuristic approach is 1588 s while the CPLEX's execution time is only 334 s on average. Hence, the proposed hybrid method still has room to improve, especially in terms of execution speed.

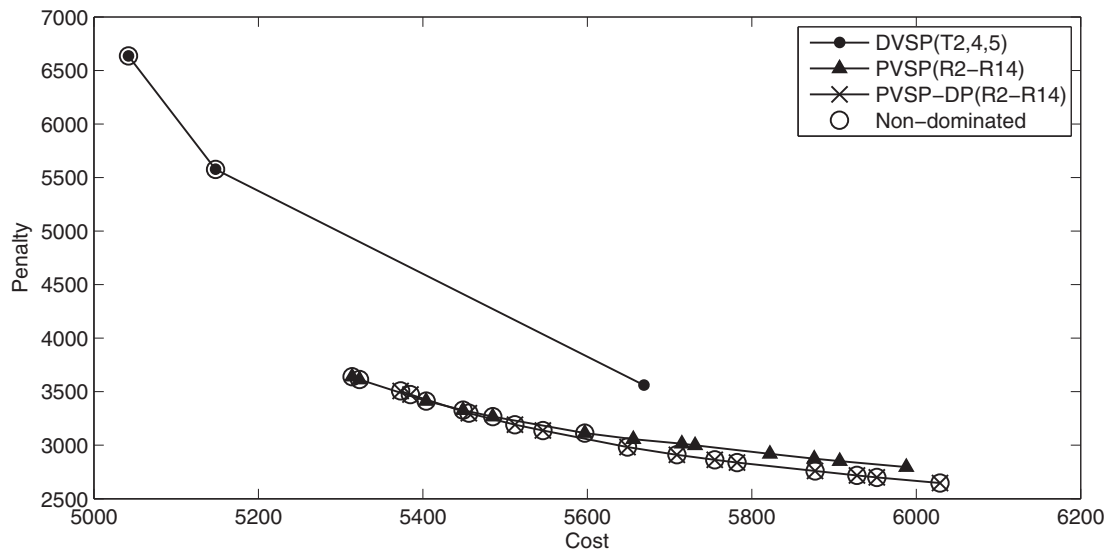*4.3.2. The PVSP model vs. the PVSP-DP model*

This section aims to compare the two probabilistic models: PVSP and PVSP-DP. Using the same α values (indexed from R1 to R15) as listed in Table 4, the heuristic approach produces 15 schedules based on the PVSP and PVSP-DP models respectively as shown in Table 6, the RPDs are computed over the best known schedule (i.e. T5's schedule in Table 3).

From Table 6, we can see that with the same α value, the fleet size of the schedules for PVSP and PVSP-DP are all the same, therefore we focus on comparing the cost and the penalty, as well as the objective value. For both DVSP and DVSP-DP models, R1's (i.e. $\alpha = 0.2$) schedules have the smallest fleet size, however the on-time performance are also the worst for each model respectively, hence the two schedules are to be removed. R15's schedules are also to be removed, where the fleet size exceeds the expectation. With the expected fleet size, R14 (i.e. $\alpha = 1.5$) corresponds to the schedule with the highest on-time performance. The RPDs of the schedules for R14 based on DVSP and DVSP-DP in terms of cost are 11.82% and 12.59%

Table 6
Experimental results of PVSP and PVSP-DP (produced by the heuristic approach).

Index	α	Fleet size		Cost				Penalty				Objective value			
		PVSP	PVSP-DP	PVSP	RPD	PVSP-DP	RPD	PVSP	RPD	PVSP-DP	RPD	PVSP	RPD	PVSP-DP	RPD
R1	0.2	27	27	5618	4.91%	5670	5.88%	5997	65.34%	5857	61.48%	41614	7.26%	41456	6.86%
R2	0.3	28	28	5314	−0.77%	5373	0.34%	3641	0.37%	3507	−3.31%	38775	−0.05%	38634	−0.42%
R3	0.4	28	28	5323	−0.60%	5385	0.56%	3614	−0.37%	3474	−4.23%	38744	−0.13%	38596	−0.52%
R4	0.5	28	28	5404	0.92%	5456	1.89%	3413	−5.90%	3299	−9.04%	38524	−0.70%	38405	−1.01%
R5	0.6	28	28	5449	1.76%	5512	2.93%	3327	−8.29%	3192	−12.00%	38439	−0.92%	38300	−1.28%
R6	0.7	28	28	5485	2.43%	5546	3.57%	3267	−9.92%	3137	−13.53%	38386	−1.06%	38251	−1.40%
R7	0.8	28	28	5597	4.52%	5649	5.49%	3113	−14.19%	2983	−17.77%	38266	−1.37%	38123	−1.73%
R8	0.9	28	28	5656	5.62%	5709	6.61%	3057	−15.73%	2911	−19.76%	38241	−1.43%	38075	−1.86%
R9	1	28	28	5715	6.72%	5755	7.47%	3014	−16.91%	2863	−21.06%	38236	−1.44%	38050	−1.92%
R10	1.1	28	28	5731	7.02%	5782	7.97%	2999	−17.33%	2838	−21.76%	38229	−1.46%	38039	−1.95%
R11	1.2	28	28	5822	8.72%	5877	9.75%	2919	−19.54%	2759	−23.95%	38200	−1.54%	38015	−2.01%
R12	1.3	28	28	5876	9.73%	5928	10.70%	2874	−20.77%	2717	−25.09%	38187	−1.57%	38004	−2.04%
R13	1.4	28	28	5907	10.31%	5952	11.15%	2851	−21.39%	2699	−25.58%	38184	−1.58%	38001	−2.05%
R14	1.5	28	28	5988	11.82%	6029	12.59%	2796	−22.92%	2647	−27.02%	38182	−1.58%	38000	−2.05%
R15	1.6	29	29	5951	11.13%	5951	11.13%	2109	−41.87%	2109	−41.87%	38114	−1.76%	38114	−1.76%
Avg.		28	28	5655	5.62%	5704	6.53%	5997	−9.96%	5857	−13.63%	41614	−0.62%	41456	−1.01%

Fig. 7. Results produced by the heuristic approach based on the DVSP, PVSP and PVSP-DP models.

respectively, in terms of penalty are −22.92% and −27.02% respectively, and in terms of the objective value are −1.58% and −2.05% respectively. These results demonstrate that both probabilistic models can produce the schedules with considerable higher on-time performance than the best known schedule based on DVSP, while the schedules' costs are larger but with the same fleet size. When considering delay propagation, the PVSP-DP model can increase the on-time performance further than the PVSP model with a little compromise on the cost.

Furthermore, in Fig. 7 the schedules with 28 vehicles are compared in consideration of the cost and penalty. Fig. 7 shows the Pareto fronts for the DVSP, PVSP and PVSP-DP models, where the non-dominated schedules for the three models are cycled.

From Fig. 7, we can see that 21 out of 29 schedules are non-dominated, among which 13 are produced based on DVSP-DP, 6 are based on PVSP and 2 are based on DVSP. Obviously, the two probabilistic models generally surpass the DVSP model. In comparison of the two probabilistic models, under most circumstances the schedules produced by the PVSD-DP model are more robust due to the consideration of delay propagation.

5. Conclusion

This paper proposes two probabilistic models, namely PVSP and PVSP-DP, for vehicle scheduling with stochastic trip times. The stochastic trip times follow the probability distributions obtained by the trip time measurement based on AVL

data in advance. A case study on real instances of Haikou bus Route 4 (HKB4) has been reported, where the trip time distribution was obtained from the AVL data collected during May to August 2010.

Experimental results on a set of problem instances derived from the HKB4 case show that both of the probabilistic models can produce the schedules with the same fleet size but considerably higher on-time performance than the best known schedule generated by the traditional deterministic VSP model (the RPDs in terms of penalty are –22.92% and –27.02%, respectively). Comparing the two probabilistic models, the one that considers delay propagation (i.e. PVSP-DP) may further increase the on-time performance, while the fleet size remains the same but with a little compromise in the operating cost. Moreover, with the aid of the probabilistic models, human scheduler can be saved from the work of determining scheduled trip times, which is non-trivial and often frustrates the scheduler.

Since the existing vehicle scheduling approaches are mostly developed to solve the traditional deterministic VSP, none of them can be directly used to solve the proposed PVSP-DP model. Therefore, this paper has developed a hybrid approach that combines search by integer linear programming and search by heuristics. Although the proposed approach has still room for further improvement, especially on the aspect of search efficiency, experiments show that it can produce the results close to those produced by CPLEX and hence can be used as a solver to the PVSP-DP model.

Acknowledgments

Reference

Amberg, B., Amberg, B., Kliewer, N., 2011. Increasing delay-tolerance of vehicle and crew schedules in public transport by sequential, partial-integrated and integrated approaches. Proced.-Soc. Behav. Sci. 20, 292–301.
Ball, M., Bodin, L., Dial, R., 1983. A matching based heuristic for scheduling mass transit crews and vehicles. Transp. Sci. 17 (1), 4–31.
Bertossi, A.A., Carraresi, P., Gallo, G., 1987. On some matching problems arising in vehicle scheduling models. Networks 17 (3), 271–281.
Bodin, L., Golden, B., Assad, A., Ball, M., 1983. Routing and scheduling of vehicles and crews: the state of the art. Comput. Oper. Res. 10 (2), 63–211.
Bunte, S., Kliewer, N., 2009. An overview on vehicle scheduling models. Public Transp. 1 (4), 299–317.
Ceder, A., 2011. Public-transport vehicle scheduling with multi vehicle type. Transp. Res. Part C 19 (3), 485–497.
Hassold, S., Ceder, A., 2014. Public transport vehicle scheduling featuring multiple vehicle types. Transp. Res. Part B 67, 129–143.
Desaulniers, G., Lavigne, J., Soumis, F., 1998. Multi-depot vehicle scheduling problems with time windows and waiting costs. Eur. J.Oper. Res. 111 (3), 479–494.
Furth, P.G., 2006. TCRP Report 113: Using Archived AVL-APC Data to Improve Transit Performance and Management. Transportation Research Board of the National Academies, Washington, DC.
Furth, P.G., Muller, T.H.J., 2007. Service reliability and optimal running time schedules. Transp. Res. Rec.: J. Transp. Res. Board 2034 (1), 55–61.
Gavish, B., Shlifer, E., 1979. An approach for solving a class of transportation scheduling problems. Eur. J. Oper. Res. 3 (2), 122–134.
Hadjar, A., Marcotte, O., Soumis, F., 2006. A branch-and-cut algorithm for the multiple depot vehicle scheduling problem. Oper. Res. 54 (1), 130–149.
Haghani, A., Banihashemi, M., 2002. Heuristic approaches for solving large-scale bus transit vehicle scheduling problem with route time constraints. Transp. Res. Part A 36 (4), 309–333.
Huisman, D., Freling, R., Wagelmans, A.P.M., 2004. A robust solution approach to the dynamic vehicle scheduling problem. Transp. Sci. 38 (4), 447–458.
Huisman, D., Freling, R., 2005. Multiple-depot integrated vehicle and crew scheduling. Transp. Sci. 39 (4), 491–502.
Ibarra-Rojas, O.J., Giesen, R., Rios-Solis, Y.A., 2014. An integrated approach for timetabling and vehicle scheduling problems to analyze the trade-off between level of service and operating costs of transit networks. Transp. Res. Part B 70, 35–46.
Kittelson, Associates, 2003. TCRP Report 100: Transit Capacity and Quality of Service Manual. Transportation Research Board, Washington DC.
Kliewer, N., Amberg, B., Amberg, B., 2011. Multiple depot vehicle and crew scheduling with time windows for scheduled trips. Public Transp. 3 (3), 213–244.
Kliewer, N., Mellouli, T., Suhl, L., 2006. A time–space network based exact optimization model for multi-depot bus scheduling. Eur. J. Oper. Res. 175 (3), 1616–1627.
Kramkowski, S., Kliewer, N., Meier, C., 2009. Heuristic methods for increasing delay-tolerance of vehicle schedules in public bus transport. In: Proceedings of the Metaheuristic International Conference VIII. Hamburg, Germany.
Muller, T.H.J., and Furth, P.G., 2000. Integrating bus service planning with analysis, operational control, and performance monitoring. ITS America 2000 Annual Meeting, Washington, D.C.
Naumann, M., Suhl, L., Kramkowski, S., 2011. A stochastic programming approach for robust vehicle scheduling in public bus transport. Proced. – Social Behav. Sci. 20 (2011), 826–835.
Salicrú, M., Fleurent, C., Armengol, J.M., 2011. Timetable-based operation in urban transport: Run-time optimisation and improvements in the operating process. Transp. Res. Part A 45 (8), 721–740.
Shen, Y., Ni, Y., 2006. A public transport scheduling system based on tabu search. Urban Public Transp. 2006 (9), 34–39.
Shen, Y., Peng, K., Chen, K., Li, J., 2013. Evolutionary crew scheduling with adaptive chromosomes. Transp. Res. Part B 56, 174–185.
Shen, Y., Xia, J., 2009. Integrated bus transit scheduling for the Beijing bus group based on a unified mode of operation. Int. Trans. Oper. Res. 16 (2), 227–242.
Shen, Y., Xu, J., Zeng, Z.Y., 2015. Public transit planning and scheduling based on AVL data in China. Int. Trans. Oper. Res. 2015, 1–23. doi:10.1111/itor.12164.
Visentini, M.S., Borenstein, D., Li, J., Mirchandani, P.B., 2014. Review of real-time vehicle schedule recovery methods in transportation services. J.Sched. 17 (6), 541–567.
Xu, J., Shen, Y., 2012. Setting scheduled trip time based on AVL data. J. Transp. Syst. Eng. Inf. Technol. 12 (5), 39–45.
Xuan, Y., Argote, J., Daganzo, C.F., 2011. Dynamic bus holding strategies for schedule reliability: optimal linear control and performance analysis. Transp. Res. Part B 45 (10), 1831–1845.
Zhao, J., Dessouky, M., Bukkapatnam, S., 2006. Optimal slack time for schedule-based transit operations. Transp. Sci. 40 (4), 529–539.