Research Article

Vehicle-Scheduling Model for Operation Based on Single-Depot

Jing Teng,¹ Shuang Jin,² Xiongfei Lai,¹ and Sijin Chen¹

¹Key Laboratory of Road and Traffic Engineering of the Ministry of Education, Tongji University,
4800 Cao'an Highway, Shanghai 201804, China

²School of Transportation Engineering, Tongji University, 4800 Cao'an Highway, Shanghai 201804, China

Correspondence should be addressed to Shuang Jin; yueluowutishuang@163.com

Received 23 January 2015; Accepted 22 February 2015

Academic Editor: Wei (David) Fan

Copyright © 2015 Jing Teng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Centralized assigning of bus running between multiple lines can save operation cost of transit agency. As more big transit terminals can serve for multiple bus lines being established, coordinating the operation of these lines' vehicles becomes more economical and perspective. This paper proposed a vehicle-scheduling model for multiple lines which share vehicle resource together and service based on the same terminal. The optimization goal is to minimize the number of vehicles while considering reducing the invalid operation time under the constraint of timetable schemes and matching time for vehicle crossing two lines. A case in Ningbo city, China, was conducted to compare the performance of the cross-line schedules with the original schedules assigning vehicles within respective lines. The optimized schedules can reduce 7.14% vehicles in need while meeting the timetable schemes of all bus lines, which indicated that the proposed model is suitable for operation practice.

1. Introduction

Vehicle-schedule is an important production plan for transit agency which assigns vehicle trips according to timetable. The quality of the vehicle-schedule is mainly shown with the number of vehicles which means that fewer vehicles used usually need lower cost. Traditionally, a vehicle only runs within a single line and could not be swapped to the other line. The average service headway of a bus line per hour usually varies in a day to meet the fluctuation of passenger demand, and the number of vehicles equipped with a bus line depends on the average departure headway in peak hour. Therefore, there are often some vehicles being surplus in off-peak hour. Comparatively, the cross-line operation mode assigning vehicles to multiple lines could save the number of vehicles and especially could be feasible to the lines whose distributions of passenger demands are staggered complementary. In addition, the cross-line scheduling can promote the efficiency of human resources and vehicle maintenance facilities. With the informatization of public transportation industry, many transit agencies have set up control centers, which have the requirement of centrally and economically dispatching vehicle resource [1]. The cross-line operation mode has a good prospect for development.

The vehicle-scheduling of cross-line operation mode is a research hotspot in applied mathematics. It could be divided into two categories: single-depot vehicle-scheduling (SDVS) and multidepot vehicle-scheduling (MDVS). The difference between them is whether the vehicle resource is assigned from single-depot to multiple lines or assigned from multiple depots to multiple lines. SDVS problem gets more attention than MDVS from the view of applicability.

Bodin and Golden in 1981 presented that SDVS problem could be solved by two-phase method when setting the optimization goal as minimizing empty vehicle trip time. However, as adding the line operation time as a constraint, this problem becomes an NP hard problem and needs heuristic algorithm to solve it [2]. In 2008, Ren et al. introduced Tabu Search algorithm into genetic algorithm to solve the optimization model with the target of minimizing passenger waiting time and operating cost. This algorithm is validated that the proposed algorithm is more efficient than standard genetic algorithm, which was ever an effective method to solve the problem of bus dispatching [3]. In 2012, Wei set the model with the goal of maximizing the utilization rate of vehicles and the constraints of schedule reliability. The model was solved through the heuristic algorithm with genetic algorithm, ant colony algorithm, and particle swarm

optimization algorithm. The result indicated that operational cost and reliability have the direct proportional relationship [4].

MDVS is more complicated than SDVS and most of the research results are theoretical. Bertossi stated that the MDVS is an NP problem. In 1981, Ceder and Stern set up timetable compensation function based on the difference of actual arrival time and timetable. This method could establish the connection of the arrival time of each vehicle and start time of its first subsequent vehicle. But this method was not suitable for solving large-scale problems [5]. In 1990, Lamatsch proposed a time-space network method to solve multiproduct matching problem [6]. In the same year, Mesquita and Paixao solved a problem of multidepot vehicle-scheduling problem as a multiproduct matching problem and proposed a network flow calculation method. The problem involved two cases: 2 depots with 250 trips and 3 depots with 200 trips [7]. In 1997, Lobel modified Forbes model and successfully solved a problem in Germany with 49 depots and 25000 runs and optimized thirteen million variables through the column generation technique of Lagrange pricing method to adjust linear slack time problem with multiple depots. However, his method did not consider the constraint of line operation time [8]. In 2009, Mao and Li solved transit vehicle scheduling through the general fixed job sorting algorithm, which could improve the utilization of vehicles. But the method increased the dwell time in the depot greatly [9]. In 2011, Naumann et al. presented a new robust stochastic programming method through adjusting the fixed slack time between two adjacent trips to reduce the total expected cost. This model enhanced the computation complexity so that it is only suitable for small- and medium-sized vehicle-scheduling problem [10].

As the enterprise's operation requires high applicability, MDVS is difficult to guarantee the reliability of implementation. Relatively, SDVS is easier to manage, especially when more lines service the same terminal, because it can reduce vehicle's invalid dwelling time for matching two trips and is convenient for the passengers to transfer through the vehicle when crossing between lines. The majority of research results widely set up nonlinear multiobject models for SDVS and adopt heuristic algorithms. In consideration of the efficiency and reliability, most of the SDVS models and algorithms cannot be applied in practice. After reviewing the previous research, this paper puts forward a method of using fixed job sorting algorithm which belongs to deterministic algorithm to solve SDVS.

2. Problem Description

Figure 1 shows the cross-line operation mode based on a terminal in this paper.

The background of this operation mode is as follows.

- (1) There are many bus terminals or intermodal transit terminals established or being planned in big cities; transit-oriented development especially based on terminals has become one tendency of urban planning in developing countries, such as China.

FIGURE 1: The cross-line operation mode based on a terminal.

- (2) Laying out bus transit network based on terminals is beneficial for fleet management and maintenance and is also convenient for transferring organization of passenger flow.
- (3) Many terminals are located among functional areas to exchange passengers. Some of the lines could connect residential zones, and some could connect commercial zones or industrial zones. The peak time of the passenger volume of these lines is staggered. The transport capabilities required for these lines are complementary.
- (4) It is much easier for crew-scheduling based on the SDVS vehicle-scheduling than based on MDVS.

Minimizing the total number of vehicles used is set as a single optimization target in this paper. Some previous researches [11] still considered another target of balancing the kilometers of all the vehicles running in a day. After the investigation of six transit agencies in China, we found that the target can be omitted because continuous rolling of the vehicles' work during a period such as one month can easily balance the kilometers of all the vehicles. Therefore, this paper did not take the target into the objective function.

The timetable schemes of each of the bus lines which set the average headway or frequency in each hour are considered as known and optimized based on passenger demands in this paper. The vehicle-scheduling results must get the trips to fit the timetable scheme of each line. Every vehicle will carry out a trip chain no matter being within one line or crossing several lines. The following constraints of the vehicle-scheduling need be confirmed.

- (1) Every trip of the timetable is executed.
- (2) Each bus can only conduct one trip at the same time.
- (3) Each trip can only be conducted by one bus.
- (4) The connecting time of two consecutive trips run by one vehicle must meet the minimal dwell time in

TABLE 1: The time of vehicle departing from the terminal in existing timetable (Line 9).

Bus number	The departure time of Line 9 in Gymnasium Station								
1	6:00 ₍₁₎	8:00 ₍₂₅₎	10:00 ₍₄₈₎	12:00 ₍₆₆₎	14:00 ₍₈₃₎	16:00 ₍₁₀₆₎	18:00 ₍₁₃₄₎	20:00 ₍₁₅₇₎	22:00 ₍₁₇₄₎
2	6:07 ₍₂₎	8:08 ₍₂₇₎	10:10 ₍₄₉₎	12:10 ₍₆₇₎	14:08 ₍₈₄₎	16:07 ₍₁₀₈₎	18:06 ₍₁₃₅₎	20:10 ₍₁₅₈₎	
3	6:14 ₍₃₎	8:16 ₍₂₉₎	10:20 ₍₅₁₎	12:20 ₍₆₉₎	14:24 ₍₈₇₎	16:21 ₍₁₁₂₎	18:18 ₍₁₃₈₎	20:20 ₍₁₆₀₎	
4	6:21 ₍₄₎	8:24 ₍₃₀₎				16:28 ₍₁₁₃₎	18:25 ₍₁₃₉₎		
5	6:28 ₍₅₎				14:16 ₍₈₆₎	16:14 ₍₁₁₀₎	18:12 ₍₁₃₇₎		
6	6:35 ₍₆₎	8:32 ₍₃₂₎	10:30 ₍₅₂₎	12:30 ₍₇₀₎	14:32 ₍₈₉₎	16:35 ₍₁₁₅₎	18:32 ₍₁₄₁₎	20:30 ₍₁₆₁₎	22:15 ₍₁₇₅₎
7	6:42 ₍₇₎	8:40 ₍₃₃₎	10:40 ₍₅₄₎	12:40 ₍₇₁₎	14:40 ₍₉₀₎	16:42 ₍₁₁₇₎	18:39 ₍₁₄₂₎	20:40 ₍₁₆₃₎	
8	6:49 ₍₈₎	8:48 ₍₃₅₎	10:50 ₍₅₅₎	12:50 ₍₇₃₎	14:48 ₍₉₁₎	16:49 ₍₁₁₈₎	18:46 ₍₁₄₃₎	20:50 ₍₁₆₄₎	22:30 ₍₁₇₇₎
9	6:56 ₍₁₀₎	8:56 ₍₃₆₎	11:00 ₍₅₇₎	13:00 ₍₇₄₎	15:04 ₍₉₄₎	17:03 ₍₁₂₁₎	19:00 ₍₁₄₆₎	21:00 ₍₁₆₅₎	22:45 ₍₁₇₉₎
10	7:03 ₍₁₂₎	9:04 ₍₃₈₎				17:24 ₍₁₂₆₎	19:21 ₍₁₅₀₎		
11	7:10 ₍₁₃₎	9:12 ₍₃₉₎	11:10 ₍₅₈₎	13:10 ₍₇₆₎	15:12 ₍₉₆₎	17:10 ₍₁₂₃₎	19:07 ₍₁₄₇₎	21:10 ₍₁₆₇₎	
12	7:17 ₍₁₅₎	9:20 ₍₄₁₎	11:20 ₍₆₀₎	13:20 ₍₇₇₎	15:20 ₍₉₇₎	17:17 ₍₁₂₄₎	19:14 ₍₁₄₈₎	21:20 ₍₁₆₈₎	
13	7:24 ₍₁₇₎				14:56 ₍₉₃₎	16:56 ₍₁₂₀₎	18:53 ₍₁₄₄₎		
14	7:30 ₍₁₈₎	9:28 ₍₄₂₎	11:30 ₍₆₁₎	13:30 ₍₇₈₎	15:28 ₍₉₉₎	17:30 ₍₁₂₇₎	19:27 ₍₁₅₁₎	21:30 ₍₁₇₀₎	
15	7:36 ₍₂₀₎	9:36 ₍₄₄₎	11:40 ₍₆₃₎	13:40 ₍₈₀₎	15:44 ₍₁₀₃₎	17:42 ₍₁₃₀₎	19:39 ₍₁₅₃₎	21:40 ₍₁₇₁₎	
16	7:42 ₍₂₁₎	9:44 ₍₄₅₎				17:48 ₍₁₃₁₎	19:46 ₍₁₅₅₎		
17	7:48 ₍₂₃₎				15:36 ₍₁₀₁₎	17:36 ₍₁₂₉₎	19:33 ₍₁₅₂₎		
18	7:54 ₍₂₄₎	9:52 ₍₄₇₎	11:50 ₍₆₄₎	13:50 ₍₈₁₎	15:52 ₍₁₀₅₎	17:54 ₍₁₃₃₎	19:53 ₍₁₅₆₎	21:50 ₍₁₇₂₎	

Note. The subscript in each cell is according to the time sequence of the timetables of Line 9 and timetables of Line 519.

the terminal which is ruled according to the basic operation works.

3. Model Building

There is a premise to be stated before proposing the model that the cross-lines are equipped and run with the same type of buses in order to ensure all vehicles can serve for different bus lines.

3.1. The Mathematical Symbols. Let $\bar{R} = \{R_j \mid j = 1, 2, \dots, n\}$ be the set of cross-lines, in which j is the serial number of bus lines and n is the number of the bus lines to be optimized in vehicle-scheduling.

Let $\bar{J} = \{J_{jt} \mid j = 1, 2, \dots, n, t = 1, 2, \dots, y_j\}$ be the set of the whole trips departed from the terminal. J_{jt} means the t th trip of Line j . y_j is the number of Line j 's trips in a day.

Let $\bar{P} = \{P_k \mid k = 1, 2, \dots, m\}$ be the set of vehicles, in which k is the serial number of vehicles and m is the total number of vehicles.

Let t_{djt} be J_{jt} 's departure time from the terminal and let t_{ajt} be J_{jt} 's arrival time (the time back the terminal), which are determined by timetables.

Let t'_{jt} and t''_{jt} be the upgoing running time and the downgoing running time.

Let tt'_{jt} and tt''_{jt} be the dwell times of vehicle in origin station and end station.

Let T_{jt} be the operational time of the trip t of the bus line j ; $T_{jt} = t'_{jt} + tt'_{jt} + t''_{jt} + tt''_{jt}$ and usually $tt'_{jt} + tt''_{jt} = 15\%T_{jt}$.

Let G_{is} be the necessary time dwelling in the terminal for a vehicle connecting two consecutive trips J_{is} and J_{jt} . G_{is} is mainly affected by the first trip J_{is} . Generally, its value ranges about $10\%T_{is} \leq G_{is} \leq 15\%T_{is}$.

3.2. Model Formation. With the above mathematical symbols, the model of vehicle-scheduling was built in this section.

(1) Decision variable $X_{is,jt}^k$ is expressed as follows:

$$X_{is,jt}^k = \begin{cases} 1, & \text{after conducting trip } J_{is}, \\ & \text{the vehicle } P_k \text{ conducts trip } J_{jt}; \\ 0, & \text{otherwise.} \end{cases} \quad (1)$$

(2) Objective function is expressed as follows:

the number of vehicles, running all of the trips, is Z :

$$Z = \left\| P_k \mid \sum_{i=1}^n \sum_{s=1}^{y_i} \sum_{j=1}^{n+1} \sum_{t=1}^{y_j} X_{is,jt}^k \geq 1, \forall k \right\|. \quad (2)$$

Among them, $X_{is,(n+1)t}^k$ expresses that vehicle would not conduct any trip task after conducting trip J_{is} . Therefore, according to the principle of minimizing the number of vehicles, the objective function could be expressed as follows:

$$\min Z = \left\| P_k \mid \sum_{i=1}^n \sum_{s=1}^{y_i} \sum_{j=1}^{n+1} \sum_{t=1}^{y_j} X_{is,jt}^k \geq 1, \forall k \right\|. \quad (3)$$

(3) A vehicle can only conduct one trip at the same time, and then a vehicle can only conduct a follow-up trip J_{jt} after conducting trip J_{is} . That is to say, there is a "value of 1" in the decision variable sequence $X_{is,jt}^k$ for the given parameter k and parameter is at most:

$$\sum_{j=1}^{n+1} \sum_{t=1}^{y_j} X_{is,jt}^k \leq 1, \quad \forall k, i, s. \quad (4)$$

TABLE 2: The time of vehicle departing from the terminal in existing timetable (Line 519).

Bus number	The departure time of Line 519 in Gymnasium Station								
1	6:50 ₍₉₎ 21:50 ₍₁₇₃₎	8:30 ₍₃₁₎	10:10 ₍₅₀₎	11:50 ₍₆₅₎	13:30 ₍₇₉₎	15:10 ₍₉₅₎	16:50 ₍₁₁₉₎	18:30 ₍₁₄₀₎	20:10 ₍₁₅₉₎
2	7:00 ₍₁₁₎	8:45 ₍₃₄₎	10:30 ₍₅₃₎	12:15 ₍₆₈₎	14:10 ₍₈₅₎	15:50 ₍₁₀₄₎			
3	7:10 ₍₁₄₎				13:50 ₍₈₂₎	15:30 ₍₁₀₀₎	17:20 ₍₁₂₅₎		
4	7:20 ₍₁₆₎	9:00 ₍₃₇₎	10:50 ₍₅₆₎	12:40 ₍₇₂₎	14:30 ₍₈₈₎	16:10 ₍₁₀₉₎			
5	7:30 ₍₁₉₎ 22:15 ₍₁₇₆₎	9:15 ₍₄₀₎				15:20 ₍₉₈₎	17:05 ₍₁₂₂₎	18:55 ₍₁₄₅₎	20:35 ₍₁₆₂₎
6	7:45 ₍₂₂₎	9:30 ₍₄₃₎	11:10 ₍₅₉₎	13:05 ₍₇₅₎	14:50 ₍₉₂₎	16:30 ₍₁₁₄₎			
7	8:00 ₍₂₆₎ 22:40 ₍₁₇₈₎	9:50 ₍₄₆₎	11:30 ₍₆₂₎			15:40 ₍₁₀₂₎	17:35 ₍₁₂₈₎	19:20 ₍₁₄₉₎	21:00 ₍₁₆₆₎
8	8:15 ₍₂₈₎ 23:05 ₍₁₈₀₎					16:00 ₍₁₀₇₎	17:50 ₍₁₃₂₎	19:45 ₍₁₅₄₎	21:25 ₍₁₆₉₎
9						16:20 ₍₁₁₁₎	18:10 ₍₁₃₆₎		
10						16:40 ₍₁₁₆₎			

Note. The subscript in each cell is according to the time sequence of the timetables of Line 9 and Line 519.

- (4) Similarly, a vehicle can only conduct a consecutive trip J_{is} before conducting trip J_{jt} . That is to say, there is a “value of 1” in the decision variable sequence $X_{is,jt}^k$ for the given parameter k and parameter jt at most:

$$\sum_{i=1}^{n+1} \sum_{s=1}^{y_i} X_{is,jt}^k \leq 1, \quad \forall k, j, t. \quad (5)$$

- (5) According to the principle that each trip can only be conducted by one vehicle, it can conclude that there is a “value of 1” in the decision variable sequence $X_{is,jt}^k$ for the given parameter J_{is} :

$$\sum_{k=1}^m \sum_{j=1}^{n+1} \sum_{t=1}^{y_j} X_{is,jt}^k = 1, \quad \forall i, s. \quad (6)$$

- (6) Let β_{kjt} be the matching function:

$$\beta_{kjt} = \begin{cases} 0, & \text{the vehicle } P_k \text{ is matched with the trip } J_{jt}; \\ 1, & \text{otherwise.} \end{cases} \quad (7)$$

Therefore, according to the principle that a vehicle should be matched with a trip, the equivalent condition of $\sum_{k=1}^m \sum_{j=1}^{n+1} \sum_{t=1}^{y_j} X_{is,jt}^k = 1$ is that the vehicle P_k is matched with the trip J_{jt} :

$$\sum_{j=1}^{n+1} \sum_{t=1}^{y_j} X_{is,jt}^k + \beta_{kjt} \leq 1, \quad \forall k, i, s. \quad (8)$$

- (7) The value of the matching function β_{kjt} is calculated as follows:

according to the principle that the time for a vehicle connecting two consecutive trips J_{is} and J_{jt} cannot

be smaller than G_{is} , we could get the formulation as follows:

$$t_{djt} X_{is,jt}^k \geq (t_{ais} + G_{is}) X_{is,jt}^k, \quad (\forall k, i, s, j, t). \quad (9)$$

If the above equation holds, then $\beta_{kjt} = 0$; otherwise, $\beta_{kjt} = 1$.

4. Algorithm Design

4.1. Fixed Job Sorting Theory. The description of fixed job sorting algorithm: let $J = \{J_j \mid j = 1, 2, \dots, n\}$ be the set of jobs, in which n is the number of jobs. And the jobs in this paper correspond to bus trips. Let $P = \{P_k \mid k = 1, 2, \dots, m\}$ be the set of processors, in which m is the number of processors. The processors in this paper correspond to bus vehicles. Each processor can process no more than one job at the same time, and each job can be processed by no more than one processor at the same time. In addition, once a job begins to be processed, it cannot be broken off until completion of the processing. Let t_{dj} be the departure time of the fixed processing and let t_{aj} be the arrival time of the fixed processing for a job. There is a mapping function of $P \rightarrow J$ between a processor and a job. That is to say, there is a subset $S_k \in J$ belonging to all $P_k \in P$. And a processor P_k could process every job belonging to S_k . The key is to find a schedule to minimize the total number of processors, and all of the jobs could be processed through the mapping relationship. According to the marking method in sorting theory, the algorithm could be denoted as follows:

$$P_m \mid \text{Fixed Job}, P \longrightarrow J \quad \text{Mapping} \mid \max z = \sum_k \sum_j Y_{kj}, \quad (10)$$

where P_m is the number of processors, $Y_{kj} = 1$ expresses that a processor P_k conducts job J_j , $Y_{kj} = 0$ expresses that a processor P_k does not conduct job J_j , and $\max z = \sum_k \sum_j Y_{kj}$

FIGURE 2: Algorithm logic diagram.

denotes the maximal number of jobs would be conducted by processors. That is to say, each job could be assigned with a processor.

4.2. Algorithm Design. The general fixed job sorting algorithm is based on first in first out (FIFO) rule to dispatch bus vehicles [9]. This paper adopts a modified fixed job sorting algorithm that the dispatching rule will assign the vehicle to conduct the trip with earlier departure time when it conforms to the requirement of minimum consecutive time in the depot, which can reduce the total operating time by compressing dwelling time in terminal.

In order to describe this algorithm further, the trip is defined as $\text{Job}(jt)$:

$$\text{Job}(jt) = \begin{cases} p, & \text{Trip } J_{jt} \text{ conducted by bus } P_p \ (p = 1, 2, \dots, m), \\ 0, & \text{Unfinished Trip.} \end{cases} \quad (11)$$

Let $\bar{P}_e = \{P_p \mid p = 1, \dots, m_e, m_e < m\}$ be the set of all dwelling buses in terminal, in which m_e refers to the amount of the dwelling buses. For each dwelling bus $P_p \in \bar{P}_e$, which can execute trip J_{jt} , the interval between the departure time of J_{jt} and P_k 's arrival time of last trip must be longer than the necessary consecutive time G_{is} . Therefore, for the next trip J_{jt} , a variable E can be defined as

$$E = t_{djt} X_{is,jt}^k - (t_{ais} + G_{is}) X_{is,jt}^k \quad (\forall k, i, s, j, t). \quad (12)$$

Here the priority of the bus to be dispatched for the next trip is determined by the value of $\text{Pro}(p)$. $\text{Pro}(p)$ is the set of the sequence numbers depending on E . $\text{Pro}(p) = 1, \dots, m_e, m_e$ maps the vehicle with the biggest value of E .

Here β_{kjt} can be explained as

$$\beta_{kjt} = \begin{cases} 0, & \text{Bus } P_k \text{ is conducting trip } J_{jt}, \\ 1, & \text{otherwise.} \end{cases} \quad (13)$$

Algorithm logic is shown as in Figure 2.

TABLE 3: The total number of trips and vehicles.

Bus line	The total number of trips	The total number of vehicles
Line 9	126	18
Line 519	54	10
Total	180	28

Algorithm process is as follows.

- (1) The departure of each bus will generate a dispatching event z_i which will be sorted by time series as $z_1 \leq z_2 \leq \dots \leq z_n$. Initializing $\overline{P}_e$, $\text{Pro}(p)$ and Counter = 0.
- (2) $z_x \in (z_1, \dots, z_n)$ is defined as the departure event; the corresponding trip is J_{jt} ; then search $P_k \in \overline{P}_e$ which conforms to the constraints below:

$$\begin{aligned} \beta_{kjt} &= 0, \\ \text{Pro}(k) &= \min \{ \text{Pro}(p) \mid k \in \overline{P}_e \}. \end{aligned} \quad (14)$$

If $P_k \in \overline{P}_e$ exists, then renew $\text{Job}(jt) = k$, $m_e = m_e - 1$, $\overline{P}_e = \overline{P}_e - \{P_k\}$. Consider $P_p \in P$ and $\text{Pro}(p) > \text{Pro}(k)$. Renew $\text{Pro}(p) = \text{Pro}(p) - 1$. Otherwise, renew

$$\begin{aligned} \text{Job}(jt) &= 0, \\ \text{Counter} &= \text{Counter} + 1. \end{aligned} \quad (15)$$

Consider $z_y \in (z_1, \dots, z_n)$, supposing $z_y + t'_{is} + tt'_{is} + t''_{is}$ as the finish event of trip J_{is} , and then

$$\begin{aligned} m_e &= m_e + 1, \\ \overline{P}_e &= \overline{P}_e + \{P_p \mid p = \text{Job}(is)\}, \\ \text{Pro}(\text{Job}(is)) &= m_e. \end{aligned} \quad (16)$$

- (3) Renew $k = k + 1$. If $k \leq n$, the process returns to step (2); otherwise, output $\text{Job}(jt)$, $jt = 1, 2, \dots, n$, Counter.

5. Case Study

5.1. Overview of Case. Two tested bus lines, Line 9 and Line 519 in Ningbo, China, are shown below, which share the identical terminal—Gymnasium Station. The layout of the bus lines is shown as in Figure 3.

The running time of Line 9 is 51 min. The normal time for vehicles dwelling at the terminal accounts for 15% of the running time and should not be less than 10%. There are totally 126 trips conducted by 18 vehicles every day and the time of vehicle departing from the terminal in existing timetable is shown in Table 1.

Similarly, the running time of Line 519 is 42 min. Therefore, the normal time for vehicles dwelling at the terminal is 8 min while the minimum dwelling time cannot be less than 5 min. There are 54 trips conducted by 10 vehicles every day

TABLE 4: The 26-vehicle operation diagram.

Bus label	Trip chain
P_1	$J(1), J(25), J(48), J(66), J(83), J(106)$
P_2	$J(2), J(27), J(49), J(68), J(85), J(105), J(132), J(155)$
P_3	$J(3), J(28), J(51), J(69), J(89), J(114), J(136)$
P_4	$J(4), J(30), J(93), J(120), J(144), J(164)$
P_5	$J(5), J(103), J(130), J(153)$
P_6	$J(7), J(33), J(98), J(122), J(145), J(175)$
P_7	$J(9), J(31), J(50), J(86), J(110), J(137), J(159), J(173)$
P_8	$J(10), J(36), J(56), J(70), J(88), J(108), J(135)$
P_9	$J(6), J(32), J(53), J(67), J(84), J(109), J(131), J(154), J(171)$
P_{10}	$J(12), J(37), J(54), J(72), J(87), J(112), J(138), J(160)$
P_{11}	$J(8), J(107), J(133), J(156), J(172)$
P_{12}	$J(14), J(35), J(55), J(75), J(92), J(113), J(139)$
P_{13}	$J(15), J(40), J(57), J(74), J(94), J(121), J(146), J(166), J(178)$
P_{14}	$J(11), J(34), J(52), J(71), J(90), J(116)$
P_{15}	$J(17), J(111), J(134), J(157), J(174)$
P_{16}	$J(13), J(41), J(60), J(77), J(97), J(125)$
P_{17}	$J(18), J(42), J(61), J(78), J(99), J(129), J(152), J(170)$
P_{18}	$J(20), J(44), J(63), J(80), J(102), J(124), J(148)$
P_{19}	$J(16), J(38), J(101), J(128), J(149), J(166)$
P_{20}	$J(21), J(45), J(64), J(82), J(100), J(126), J(150), J(168)$
P_{21}	$J(23), J(46), J(62), J(77), J(96), J(123), J(147), J(167)$
P_{22}	$J(19), J(39), J(59), J(73), J(91), J(119), J(140), J(158), J(176)$
P_{23}	$J(24), J(47), J(65), J(79), J(95), J(118), J(143)$
P_{24}	$J(26), J(115), J(141), J(161), J(177)$
P_{25}	$J(22), J(43), J(58), J(81), J(104), J(127), J(151), J(169), J(180)$
P_{26}	$J(29), J(117), J(153), J(163), J(179)$

and the time of vehicle departing from the terminal in the existing timetable is shown in Table 2.

The total trips and vehicles of two lines are shown in Table 3.

5.2. The Optimized Vehicle-Schedule. With the modified fixed job sorting algorithm, the cross-line vehicle-schedule can be quickly gotten and all the trips are burdened with 26 vehicles. Compared with the operation mode of single line, the method could reduce 2 vehicles which occupy 7.1% of the existing vehicle number. Table 4 shows the trip chains of the 26 vehicles in a day. Figure 4 is the operation diagram of the two bus lines under cross-line operation mode.

Table 5 shows the optimized cross-line vehicle-schedule in Gymnasium Station. It is known that during the transition phase from peak period to off-peak period, the transport capacity offered in peak period will be cut away to fit with off-peak transport demand. With the modified fixed job sorting algorithm, some vehicles could get a long nonservice

TABLE 5: The optimized cross-line vehicle-schedule of the Gymnasium Station terminal.

Bus	D.T.1	A.T.1	S.T.1	D.T.2	A.T.2	S.T.2	D.T.3	A.T.3	S.T.3	D.T.4	A.T.4	S.T.4	D.T.5	A.T.5	S.T.5	D.T.6	A.T.6	S.T.6	D.T.7	A.T.7	S.T.7	D.T.7	A.T.7	S.T.7	D.T.7	A.T.7
7	6:50	8:22	8	8:30	10:02	8	10:10	11:42	Nonservice time	Nonservice time	Nonservice time	14:16	16:07	7	16:14	18:05	7	18:12	20:03	7	20:10	21:42	8	21:50	23:22	
9	6:35	8:26	6	8:32	10:23	7	10:30	12:02				8	12:10	14:01	7	14:08	15:59	11	16:10	17:42	6	17:48	19:39	6	19:45	21:17
11	6:49	8:40						Nonservice time								16:00	17:32	22	17:54	19:45	8	19:53	21:44	6	21:50	23:41
22	7:30	9:02	10	9:12	11:03	7	11:10	12:42	8	12:50	14:41	7	14:48	16:39	11	16:50	18:22	8	18:30	20:02	8	20:10	22:01	14	22:15	23:47
15	7:24	9:15						Nonservice time								16:20	17:52	8	18:00	19:51	9	20:00	21:51	9	22:00	23:51
6	6:42	8:33	7	8:40	10:31			Nonservice time					15:20	16:52	13	17:05	18:37	18	18:55	20:27	8	20:35	22:07	8	22:15	0:06
13	7:17	9:08	7	9:15	10:47	13	11:00	12:51	9	13:00	14:51	13	15:04	16:55	8	17:03	18:54	6	19:00	20:51	9	21:00	22:32	8	22:40	0:12
24	8:00	9:32						Nonservice time								16:35	18:26	6	18:32	20:23	7	20:30	22:21	9	22:30	0:21
26	8:16	10:07						Nonservice time								16:42	18:33	6	18:39	20:30	10	20:40	22:31	14	22:45	0:36
25	7:45	9:17	13	9:30	11:02	8	11:10	13:01	41	13:50	15:41	9	15:50	17:22	8	17:30	19:21	6	19:27	21:18	7	21:25	22:57	8	23:05	0:37
2	6:07	7:58	10	8:08	9:59	11	10:10	12:01	14	12:15	13:47	13	14:10	15:42	10	15:52	17:43	7	17:50	19:22	24	19:46	21:37	Nonservice time		
10	7:03	8:54	6	9:00	10:32	8	10:40	12:31	9	12:40	14:12	12	14:24	16:15	6	16:21	18:12	6	18:18	20:09	11	20:20	22:11	Nonservice time		
4	6:21	8:12	12	8:24	10:15			Nonservice time					14:56	16:47	9	16:56	18:47	6	18:53	20:44	6	20:50	22:41	Nonservice time		
19	7:20	8:52	12	9:04	10:55			Nonservice time					15:36	17:27	8	17:35	19:07	13	19:20	20:52	8	21:00	22:51	Nonservice time		
21	7:48	9:39	11	9:50	11:22	8	11:30	13:02	8	13:10	15:01	11	15:12	17:03	7	17:10	19:01	6	19:07	20:58	12	21:10	23:01	Nonservice time		
20	7:42	9:33	11	9:44	11:35	15	11:50	13:41	9	13:50	15:22	8	15:30	17:02	22	17:24	19:15	6	19:21	21:12	8	21:20	23:11	Nonservice time		
17	7:30	9:21	7	9:28	11:19	11	11:30	13:21	9	13:30	15:21	7	15:28	17:19	17	17:36	19:27	6	19:33	21:24	6	21:30	23:21	Nonservice time		
3	6:14	8:05	10	8:15	9:47	33	10:20	12:11	9	12:20	14:11	13	14:32	16:23	7	16:30	18:02	8	18:10	19:42	Nonservice time					
8	6:56	8:47	9	8:56	10:47	13	10:50	12:22	8	12:30	14:21	9	14:30	16:02	5	16:07	17:58	8	18:06	19:57	Nonservice time					
12	7:10	8:42	8	8:48	10:39	11	10:50	12:41	24	13:05	14:37	13	14:50	16:22	6	16:28	18:19	6	18:25	20:16	Nonservice time					
23	7:54	9:45	7	9:52	11:43	7	11:50	13:22	8	13:30	15:02	8	15:10	16:42	7	16:49	18:40	6	18:46	20:37	Nonservice time					
18	7:36	9:27	9	9:36	11:27	13	11:40	13:31	9	13:40	15:31	9	15:40	17:12	5	17:17	19:08	6	19:14	21:05	Nonservice time					
5	6:28	8:19					Nonservice time						15:44	17:35	7	17:42	19:33	6	19:39	21:30	Nonservice time					
1	6:00	7:51	9	8:00	9:51	9	10:00	11:51	9	12:00	13:51	9	14:00	15:51	9	16:00	17:51				Nonservice time					
14	7:00	8:32	13	8:45	10:17	13	10:30	12:21	19	12:40	14:31	9	14:40	16:31	9	16:40	18:12				Nonservice time					
16	7:10	9:01	19	9:20	11:11	9	11:20	13:11	9	13:20	15:11	9	15:20	17:11	9	17:20	18:52				Nonservice time					

D.T., A.T., and S.T. are the abbreviations of departure time, arrival time, and dwelling time; the unit of S.T. is min.

TABLE 6: The general cross-line vehicle-schedule based on first in first out (FIFO) rule in Gymnasium Station.

Bus	D.T.1	A.T.1	S.T.1	D.T.2	A.T.2	S.T.2	D.T.3	A.T.3	S.T.3	D.T.4	A.T.4	S.T.4	D.T.5	A.T.5	S.T.5	D.T.6	A.T.6	S.T.6	D.T.7	A.T.7
1	6:00	7:51	9	8:00	9:51	49	10:40	12:31	79	13:50	15:22	38	16:00	17:51	9	18:00	19:51	39	20:30	22:21
2	6:07	7:58	10	8:08	9:59	51	10:50	12:41	69	13:50	15:41	26	16:07	17:58	12	18:10	19:42	28	20:10	21:42
3	6:14	8:05	10	8:15	9:47	43	10:30	12:21	69	13:30	15:21	31	15:52	17:43	11	17:54	19:45	35	20:20	22:11
4	6:21	8:12	12	8:24	10:15	45	11:00	12:51	77	14:08	15:59	21	16:20	17:52	14	18:06	19:57	38	20:35	22:07
5	6:28	8:19	11	8:30	10:02	48	10:50	12:22	78	13:40	15:31	29	16:00	17:32	16	17:48	19:39	31	20:10	22:01
6	6:35	8:26	14	8:40	10:31	49	11:20	13:11	73	14:24	16:15	15	16:30	18:02	10	18:12	20:03	47	20:50	22:41
7	6:42	8:33	15	8:48	10:39	51	11:30	13:21	69	14:30	16:02	19	16:21	18:12	13	18:25	20:16	44	21:00	22:32
8	6:49	8:40	16	8:56	10:47	53	11:40	13:31	69	14:40	16:31	11	16:42	18:33	20	18:53	20:44	56	21:40	23:31
9	6:50	8:22	10	8:32	10:23	47	11:10	13:01	69	14:10	15:42	28	16:10	17:42	8	17:50	19:22	38	20:00	21:51
10	6:56	8:47	17	9:04	10:55	55	11:50	13:41	67	14:48	16:39	10	16:49	18:40	20	19:00	20:51	59	21:50	23:41
11	7:00	8:32	13	8:45	10:17	53	11:10	12:42	78	14:00	15:51	23	16:14	18:05	13	18:18	20:09	51	21:00	22:51
12	7:03	8:54	21	9:15	10:47	63	11:50	13:22	70	14:32	16:23	17	16:40	18:12	18	18:30	20:02	38	20:40	22:31
13	7:10	9:01	19	9:20	11:11	64	12:15	13:47	63	14:50	16:22	13	16:35	18:26	20	18:46	20:37	53	21:30	23:21
14	7:10	8:42	18	9:00	10:32	58	11:30	13:02	74	14:16	16:07	21	16:28	18:19	13	18:32	20:23	47	21:10	23:01
15	7:17	9:08	22	9:30	11:02	58	12:00	13:51	65	14:56	16:47	9	16:56	18:47	20	19:07	20:58	62	22:00	23:51
16	7:20	8:52	20	9:12	11:03	67	12:10	14:01	63	15:04	16:55	10	17:05	18:37	18	18:55	20:27	53	21:20	23:11
17	7:24	9:15	21	9:36	11:27	73	12:40	14:12	60	15:12	17:03	14	17:17	19:08	25	19:33	21:24	81	22:45	0:36
18	7:30	9:21	29	9:50	11:22	68	12:30	14:21	59	15:20	17:11	9	17:20	18:52	22	19:14	21:05	70	22:15	0:06
19	7:30	9:02	26	9:28	11:19	61	12:20	14:11	59	15:10	16:42	8	16:50	18:22	17	18:39	20:30	55	21:25	22:57
20	7:36	9:27	25	9:52	11:43	77	13:00	14:51	45	15:36	17:27	9	17:36	19:27	19	19:46	21:37	<i>Non-service time</i>		
21	7:42	9:33	37	10:10	12:01	69	13:10	15:01	39	15:40	17:12	12	17:24	19:15	24	19:39	21:30	95	23:05	0:37
22	7:45	9:17	27	9:44	11:35	65	12:40	14:31	49	15:20	16:52	11	17:03	18:54	26	19:20	20:52	58	21:50	23:22
23	7:48	9:39	31	10:10	11:42	68	12:50	14:41	49	15:30	17:02	8	17:10	19:01	20	19:21	21:12	63	22:15	23:47
24	7:54	9:45	35	10:20	12:11	79	13:30	15:02	42	15:44	17:35	7	17:42	19:33	20	19:53	21:44	<i>Non-service time</i>		
25	8:00	9:32	28	10:00	11:51	74	13:05	14:37	51	15:28	17:19	11	17:30	19:21	24	19:45	21:17	73	22:30	0:21
26	8:16	10:07	23	10:30	12:02	78	13:20	15:11	39	15:50	17:22	13	17:35	19:07	20	19:27	21:18	82	22:40	0:12

D.T., A.T., and S.T. are the abbreviations of departure time, arrival time, and dwelling time; the unit of S.T. is min.

FIGURE 3: The layout of Line 9 and Line 519 in Ningbo city, China.

FIGURE 4: Vehicle operation diagram (all kinds of vehicles are represented by different colors).

interval (the bold italic font parts in Table 5) and others continue operating efficiently, which is beneficial for making the work plan of repairing and maintaining vehicles while rolling vehicles to execute the vehicle schedule. During the long nonservice interval, vehicles will exit operation and the bold italic font parts do not belong to the operating time, so the modified method can reduce the total operating time of the vehicles. In this case, through optimization, the total dwelling time at the terminal of 26 buses conducting 180 round trips is 1409 min. The average interval between two consecutive trips is 7.9 min.

In comparison, the general fixed job sorting algorithm selects the bus vehicle to fit a connective trip based on first in first out (FIFO) rule. During the transition phase from peak period to off-peak period the vehicles that arrived earlier have to continue conducting the subsequent trips and the total vehicles' extended dwelling time in terminal cannot be concentrated on part vehicles and then cannot be utilized for vehicle maintenance or other purposes, so the time is idle but still belongs to operating time. Table 6 shows the general cross-line vehicle-schedule based on first in first out (FIFO) rule in Gymnasium Station.

In this case, with the general fixed job sorting algorithm, the total number of vehicles optimized is still 26 buses, but the total dwelling time at the terminal is 5880 min and the average interval between two consecutive trips is 32.7 min. That is the reason why we modify the general fixed job sorting algorithm. Figure 5 shows the comparison of two circumstances.

FIGURE 5: The comparison of the average intervals between two consecutive trips of two circumstances.

6. Conclusion

This paper studied a model to optimize cross-line vehicle scheduling for single-depot, whose optimization goal is minimizing the total number of vehicles used. Setting the sole goal avoids the complexity and uncertainty of solving the multiobjective problem and still keeps availability. The modified fixed job sorting algorithm is designed, which can not only get the deterministic optimum solution to reduce total vehicle used but also save the vehicles' operating time. The algorithm can be easily written into computer program and the computation time is short, which is feasible for application. In fact, the vehicle-scheduling method would not be limited in bus transit operation and could be generalized to the similar scenario of logistics transport.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported by the National Natural Science Foundation of China (Grant no. 61174185).

References

- [1] J. Teng and X. G. Yang, "The study of transit coordination dispatching mode for big cities," in *Public Transport and Urban Development Studies and Practices*, pp. 20–28, 2006, (Chinese).
- [2] L. Bodin and B. Golden, "Classification in vehicle routing and scheduling," *Networks*, vol. 11, no. 2, pp. 97–108, 1981.
- [3] C. X. Ren, Y. J. Xun, and C. C. Yin, "Research on vehicle scheduling based on genetic tabu search algorithm," *Journal of Shandong University of Science and Technology (Natural Science Edition)*, pp. 53–56, 2008.
- [4] M. Wei, *The model and algorithm of regional vehicle scheduling under uncertain information environment [Ph.D. thesis]*, South China University of Technology, Guangzhou, China, 2012, (Chinese).

- [5] A. Ceder and H. I. Stern, "Deficit function bus scheduling with deadheading trip insertion for fleet size reduction," *Transportation Science*, vol. 15, no. 4, pp. 338–363, 1981.
- [6] A. Lamatsch, "An approach to vehicle scheduling with depot capacity constraints," in *Computer-Aided Transit Scheduling: Proceedings of the 5th International Workshop on Computer-Aided Scheduling of Public Transport held in Montréal, Canada, August 19–23, 1990*, vol. 386 of *Lecture Notes in Economics and Mathematical Systems*, pp. 181–195, Springer, Berlin, Germany, 1992.
- [7] M. Mesquita and J. Paixao, "Multiple depot vehicle scheduling problem: a new heuristic based on quasi-assignment algorithm," in *Proceedings of the 5th International Workshop on Computer-aided Scheduling of Public Transit*, Montreal, Canada, 1990.
- [8] A. Lobel, "Solving large-scale multiple-depot scheduling problems," in *Proceedings of the 7th International Conference on Computer-Aided Scheduling of Public Transit*, Cambridge, Mass, USA, 1997.
- [9] L. Mao and W. Q. Li, "Research on transit vehicle scheduling model and its algorithm," *Journal of Transportation Systems Engineering and Information*, vol. 7, no. 3, pp. 64–68, 2009 (Chinese).
- [10] M. Naumann, L. Suhl, and S. Kramkowski, "A stochastic programming approach for robust vehicle scheduling in public bus transport," *Procedia—Social and Behavioral Sciences*, vol. 20, pp. 826–835, 2011.
- [11] B. Z. Lou, *Research on optimization algorithm of vehicle scheduling based on deficit function [M.S. thesis]*, Beijing Jiaotong University, Beijing, China, 2011, (Chinese).