

Research Article

A MultiObjective Optimization Approach for Integrated Timetabling and Vehicle Scheduling with Uncertainty

Yindong Shen ¹, Wenliang Xie,¹ and Jingpeng Li²

¹Key Laboratory of Image Processing and Intelligent Control, School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan 430074, China

²Division of Computing Science and Mathematics, University of Stirling, Stirling, UK

Correspondence should be addressed to Yindong Shen; yindong@hust.edu.cn

Received 20 May 2021; Revised 6 July 2021; Accepted 26 July 2021; Published 5 August 2021

Academic Editor: Kun An

Copyright © 2021 Yindong Shen 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.

The timetabling problem (TTP) and vehicle scheduling problem (VSP) are two indispensable problems in public transit planning process. They used to be solved in sequence; hence, optimality of resulting solutions is compromised. To get better results, some integrated approaches emerge to solve the TTP and VSP as an integrated problem. In the existing integrated approaches, the passenger comfort on bus and the uncertainty in the real world are rarely considered. To provide better service for passengers and enhance the robustness of the schedule to be compiled, we study the integrated optimization of TTP and VSP with uncertainty. In this paper, a novel multiobjective optimization approach with the objectives of minimizing the passenger travel cost, the vehicle scheduling cost, and the incompatible trip-link cost is proposed. Meanwhile, a multiobjective hybrid algorithm, which is a combination of the self-adjust genetic algorithm (SGA), large neighborhood search (LNS) algorithm, and Pareto separation operator (PSO), is applied to solve the integrated optimization problem. The experimental results show that the approach outperforms existing approaches in terms of service level and robustness.

1. Introduction

The public transit planning process is usually divided in sequential steps: network route design, timetabling [1], vehicle scheduling [2], and crew scheduling and rostering [3]. The output of each activity positioned higher in the sequence becomes an important input for lower-level decisions [4]. Timetabling and vehicle scheduling are two important activities in the transit operational planning. The timetabling problem (TTP) is to find the best dispatching policy for transit vehicles with the objectives of minimizing passenger waiting time at stops and minimizing the headway unevenness governed by the headway smoothing rule [5]. These objectives refer to the service level provided to passengers. The vehicle scheduling problem (VSP) is concerned with determining the optimal allocation of vehicles to carry out all the trips in a given transit timetable. The objective is to minimize the operation cost related with the usage of vehicles and fuel consumption [6]. The TTP and VSP are

usually solved individually in sequence. This may lead to a compromised solution for the optimization problem of timetabling and vehicle scheduling as a whole [7]. In order to meet the passenger demand with high social benefits and lower operators' costs, it is necessary to coordinate the timetabling and vehicle scheduling. However, building a proper model and developing an integrated approach are challenging.

The mostly used integrated approach to the TTP and VSP is the recursive approach. The basic idea can be described as follows: (1) generate an initial timetable based on passenger demand; (2) determine the minimum number of vehicles that is needed to execute the timetable; (3) change the trip chain to reduce the vehicle scheduling cost; and (4) adjust the initial timetable and return to step (2) until the generated vehicle schedule is acceptable [8]. For example, Ceder [9] proposed a solution approach to combine timetabling and vehicle scheduling so as to improve the correspondence of vehicle departure times with passenger

demand while minimizing the fleet size required. The author stated that the proposed approach could obtain efficient schedules to the test instances from both the passenger and operator perspectives. Ceder [10] proposed a method to construct timetables based on even-headways and even-load with different vehicle types. The timetable was used as an input to solve the vehicle scheduling problem using the deficit function theory. Schmid and Ehmke [11] developed hybrid mathematical models and solution approaches to solve the vehicle scheduling problem with time windows as well as balanced departure times based on the research of Klierer et al. [12], which discussed the multiple-depot, multiple-vehicle-type bus VSP while using time-space-based instead of connection-based networks for modeling.

The recursive approach is hard to find the global optimum solution. In recent years, the approaches to build an integrated model of TTP and VSP have been widely studied. For instance, Petersen et al. [13] formulated a partial integrated model of TTP and VSP, in which the objectives were to minimize the costs of vehicles usage and passenger transfers. Meanwhile, a large neighborhood search meta-heuristic approach was proposed. Carosi et al. [14] studied the integrated timetabling and vehicle scheduling and proposed a mixed-integer linear programming (MILP) multicommodity flow-type model to obtain the vehicle schedule with desired headways and minimum deviation from the ideal frequency of service. Yue et al. [15] proposed an optimization methodology that integrated both train timetabling and rolling stock scheduling based on time-dependent passenger flow demands and employed a simulated-annealing (SA)-based heuristic algorithm to solve the proposed model.

Obviously, the integrated problem has multiple objectives, which conflict with each other. More and more researchers tend to establish multiobjective models and build intelligent optimization algorithms for the integrated problem. For example, Ibarra-Rojas et al. [16] proposed a biobjective optimization problem to jointly solve the single-depot VSP and the synchronization bus timetabling problem in which time windows for departure times were considered and the passenger demand were assumed to be constant. The objectives were to maximize the number of passengers benefited by well-timed transfers and to minimize the fleet size. Teng and Chen [17] focused on the TTP and VSP for electric buses and developed a multiobjective optimization model under single-line mode. The objectives included smoothing the vehicle departure intervals and minimizing the number of vehicles and total charging costs. A multiobjective particle swarm optimization (MOPSO) algorithm was developed to get the Pareto optimal solution set. Wang et al. [18] studied the integration of train scheduling and rolling stock circulation planning under time-varying passenger demand for an urban rail transit line. A multiobjective approximated mixed-integer linear programming approach was presented to provide better passenger services and reduce the operational complexity and costs of the scheduling.

Table 1 summarizes the relevant studies on the integration of TTP and VSP, in terms of the approach type of

integration, objectives, solution method, and data type. Among the existing studies on the integrated problem, the passenger crowding on the bus is rarely considered. The timetable obtained without considering crowding on the bus might bring passengers unsatisfied or uncomfortable and increase the cost of the transport system. Moreover, these researches do not consider the uncertainty of the passenger flow and trip times, which exists commonly in the real-world bus operation. Evaluating the passenger travel cost that consists of waiting cost and comfort cost would be inaccurate regardless of the stochastic passenger flow. Furthermore, when the stochastic trip times are employed, some infeasible trip links in the traditional VSP with fixed and tight trip times might become compatible, and the resulting schedule might be more robust (i.e., with high on-time performance) [19]. Considering the stochastic passenger flow and trip time, the integrated problem would be more complex and hard to optimize. Therefore, to have better performance, an improved optimization strategy for the integrated problem with uncertainty should be proposed.

To enhance the passenger satisfaction and the robustness of the vehicle schedule generated, we proposed a multiobjective model of the integrated TTP and VSP with stochastic passenger flow and trip times. And a hybrid algorithm is applied to optimize the multiobjective model. The remainder of the paper is structured as follows: Section 2 proposes a multiobjective optimization model for the integrated TTP and VSP, which aims at lowering the passenger travel cost, vehicle operating cost, and incompatible trip-link chain cost while satisfying the passenger and bus corporation demands. Section 3 develops a solution method to solve the integrated problem of timetabling and vehicle scheduling. Section 4 displays experimental results. Some concluding remarks and possible future research directions are given in Section 5.

2. Mathematical Model

The aim of the coordinated study about timetabling and vehicle scheduling is the assignment of departure intervals and the allocation of a fleet of vehicles to carry out all the trips in the timetables minimizing the costs at the same time. In this section, an integrated multiobjective optimization model of timetabling and vehicle scheduling is presented to solve the problem efficiently. This model considers the passenger waiting cost and comfort cost and scheduling scheme costs with the stochastic passenger flow and trip time.

2.1. Problem Description

2.1.1. Solution Structure of the Integrated Problem. The integrated problem of timetabling and vehicle scheduling is concerned with the generation of a timetable and a vehicle schedule. Considering the departure intervals are different in different time periods, the whole day trips should be divided into different periods. Hence, a timetable contains a set of headways within each time period. The departure time of each trip could be calculated based on the set of headways in

TABLE 1: Summary of relevant studies on the integration of TTP and VSP.

Approach type	Publication	Objectives	Solution method	Data type
Recursive approach	Ceder [9]	Maximize the correspondence of vehicle departure times, the fleet size	Deficit function approach	Fixed passenger load at each stop, fixed trip time
	Ceder [4]	Even-headways and even-load with different vehicle types, minimize vehicle scheduling cost	Heuristics	Fixed passenger load at each stop, fixed trip time
	Schmid and Ehmke [11]	Minimize costs of operation, maximize quality of timetables	Hybrid metaheuristic	Fixed time windows, fixed trip time
Integrated model	Petersen et al. [13]	Minimize the costs of vehicles usage and passenger transfers	LNS approach	Fixed passenger volume estimates, fixed trip time
	Carosi et al. [14]	Minimum deviation from the ideal frequency of service, vehicle schedule cost	Diving-type approach	Changeable time window, fixed trip time
	Yue et al. [15]	Minimizing infeasible trains, waiting times for passengers	SA algorithm	Fixed passenger volume, fixed trip time
Multi-objective model	Ibarra-Rojas et al. [16]	Maximize the number of passengers benefited, minimize the fleet size	ϵ -constraint method	Fixed passenger volume estimates, fixed trip time
	Teng and Chen [17]	Smooth the vehicle departure intervals, minimize the number of vehicles and total charging costs	MOPSO algorithm	Fixed trip time
	Wang et al. [18]	Minimize the load factor variation and the headway variation of trains, the number of entering and existing depot operations	MINLP approach	Fixed passenger traveling rates, fixed trip time limits

the timetable. An example of the timetable based on headway is presented as Figure 1. Based on the timetable, the vehicle schedule is acquired with a network flow model. The network flow model is shown in Figure 2, where the letter (A or B) in the cycle before a solid arrow denotes the departure time/point of a trip, while the letter in the cycle after a solid arrow denotes the arrival time/point of a trip. 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. Each trip has to be assigned to exactly one vehicle. The connection of two consecutive trips in a vehicle is called a trip-link arc.

A solution that contains the set of headway and allocation of the vehicle trip-link chain can refer to the scheduling scheme of the integrated problem. Suppose there are f time periods, the solution structure of the integrated problem is shown in Figure 3, where T_0 is the beginning of the timetable and N is the fleet size of the vehicle schedule.

2.1.2. Stochastic Passenger Flow and Trip Time Problem.

The passenger boarding and alighting flow would be different under different timetables. A small example of the changeable passenger flow is presented in Figure 4. From the illustration, we can observe that the passenger flow varies at same time under different timetables, and the distribution of passenger flow follows some certain rules. Therefore, evaluating the cost based on a certain value of the passenger flow would be inaccurate. To estimate the passenger flow under different timetables feasibly, the passenger flow may be calculated as an integral of the distribution.

Considering the stochastic trip time, the compatibility of any pair of trips becomes uncertain, and the compatibility probability is defined to determine the existence of the arcs between trips. A compatibility probability of two trips i and j can be defined as illustrated in Figure 5, where the curve denotes the arrival time distribution $f_i^e(t)$

of trip i ; $T_i^{e,car}$ and $T_i^{e,lat}$ denote the earliest and latest arrival times of trip i , respectively; T_j denotes the scheduled departure time of trip j ; and DH_{ij} denotes the deadhead time [19]. Different from the VSP with certain trip time, the compatibility probability of the pair of trips is defined as an integral of the probability distribution. Evaluating the cost of trip-link arc with uncertainty is more complicated since the incompatibility probability of the pair of trips should be penalized. Dividing the cost of VSP into the vehicle scheduling cost and the incompatible trip-link penalty cost would simplify the evaluating of the cost and make the influence of the stochastic trip time more obvious.

2.1.3. Multiobjective Optimization Problem.

As mentioned before, there are three optimization objectives in the integrated problem, which are minimizing the passenger travel cost, minimizing the vehicle scheduling cost, and minimizing the incompatible trip-link costs. The passenger travel cost reflects the passenger waiting time spent on each stop and passenger crowding kilometers on the bus. The costs of the available vehicles and the links between trips consist of the cost of vehicle scheduling and reflect the vehicle cost and fuel consumption. The incompatible trip-link cost penalizes the infeasible trip link in the trip chain, which is attributed to the stochastic trip time. Obviously, these optimization objectives conflict with each other. A shorter headway would reduce the passenger cost but require more bus vehicles, which would increase the vehicle scheduling cost. A lower incompatibility trip-link penalty would require a longer headway and shorter trip-link chain, which would increase the passenger cost and vehicle scheduling cost. Therefore, establishing a multiobjective optimization model is a beneficial way to solve the integrated problem. The relationship between the objectives in the integrated problem is shown as Figure 6.

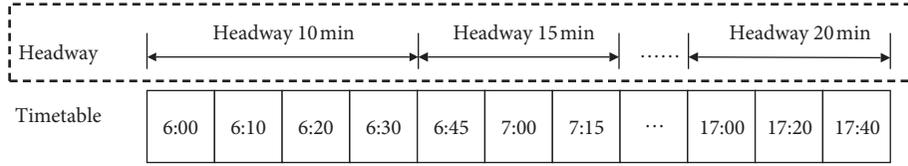


FIGURE 1: The example of the timetable based on headway.

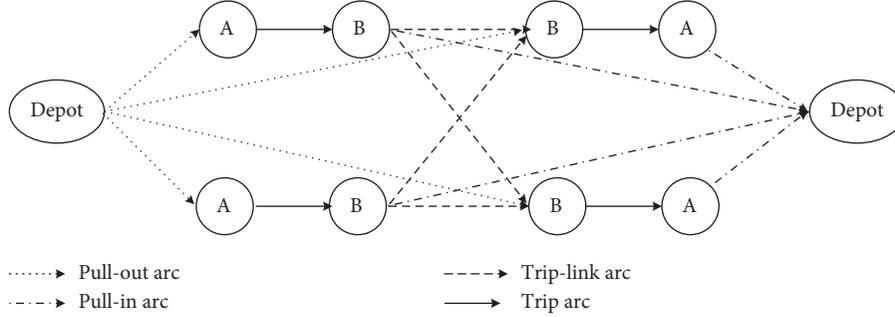


FIGURE 2: A network flow model for the VSP (Figure 2 is reproduced from [19]).

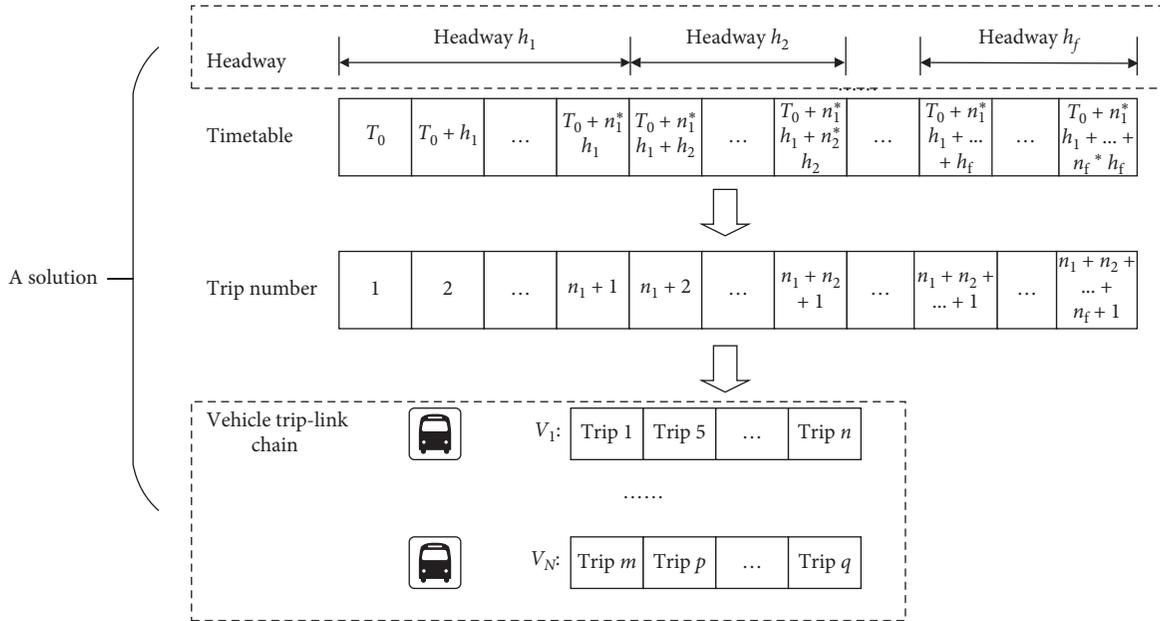


FIGURE 3: Schematic diagram of solution structure.

2.2. A Multiobjective Optimization Model for the Integrated Problem. To formulate a mathematical model for the integrated timetabling and vehicle scheduling problem, some notations are provided in Tables 2 and 3.

The objectives of the multiobjective optimization model are to minimize the passenger travel cost, the bus vehicle scheduling cost, and the incompatible trip-link cost. The solution of the integrated model would optimize the objectives and satisfy the constraints at the same time. In the integrated model, the decision variables are departure interval and arc link, which refer to the headway and vehicle trip-link chain, respectively. The integrated model can be expressed as

$$\begin{aligned}
 \min Z &= (Z_1, Z_2, Z_3), \\
 \text{where } Z_1 &= \sum_{s=1}^S \sum_{n=1}^N (t_{s,n} - t_{s,n-1}) \cdot \int_{t_{s,n-1}}^{t_{s,n}} f_s^b(t) dt \cdot Q_s^b \\
 &+ \alpha \cdot \sum_{n=1}^N \sum_{s=1}^S (t_{s+1,n} - t_{s,n}) \cdot F_c(n, s), \\
 Z_2 &= \sum_{(d,j) \in P} C_{veh} \cdot x_{dj} + \sum_{(i,j) \in A} C_{ij} \cdot x_{ij}, \\
 Z_3 &= \sum_{(i,j) \in R} p_{ij} \cdot x_{ij},
 \end{aligned}$$

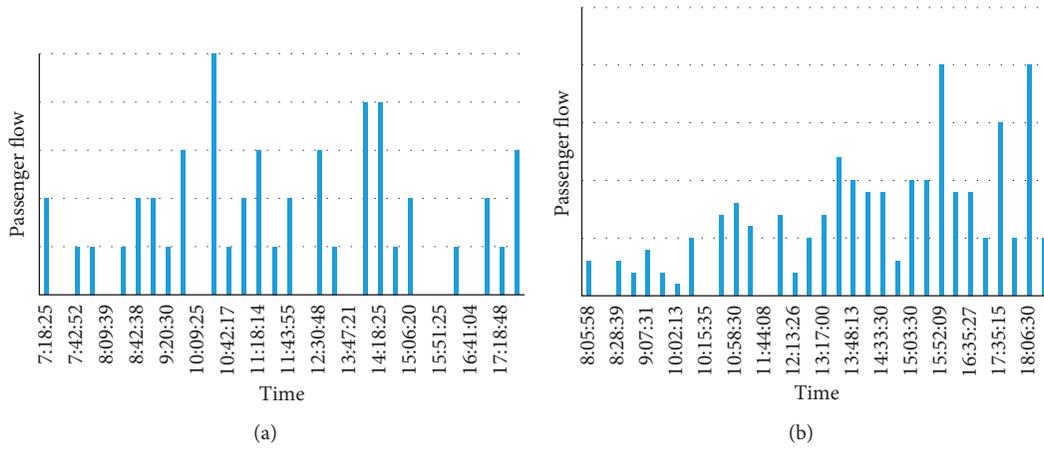


FIGURE 4: An example of the passenger flow of same stop under different timetables.

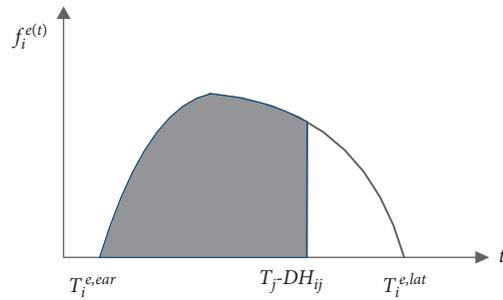


FIGURE 5: Compatibility probability of any two trips i and j (Figure 5 is reproduced from [19]).

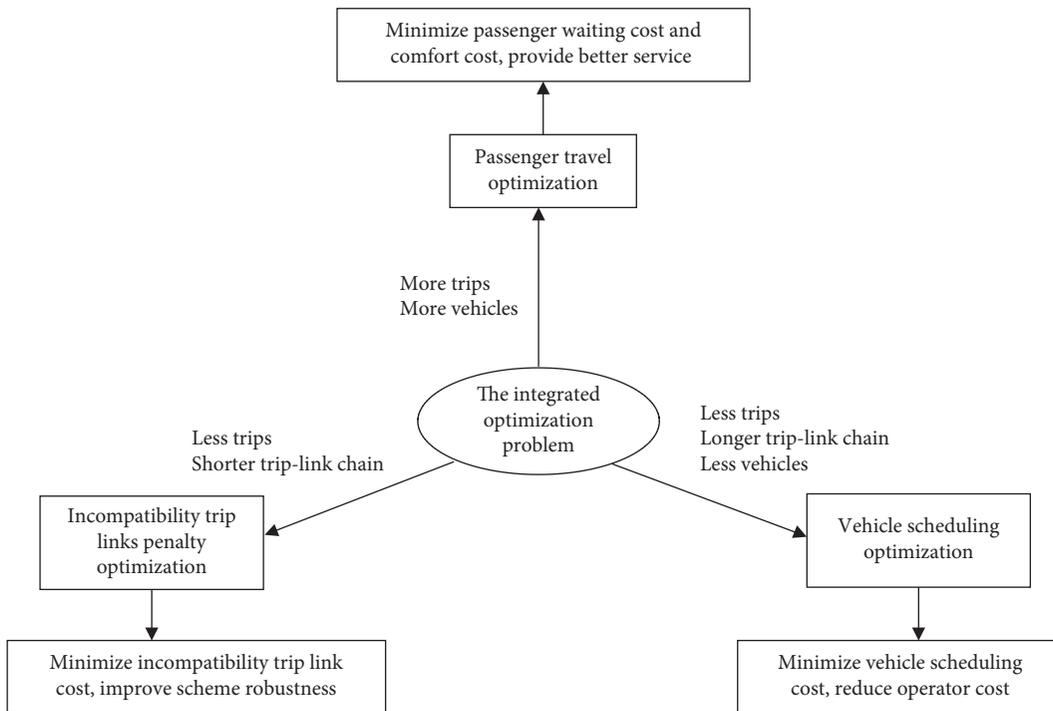


FIGURE 6: The conflict objectives of the integrated model.

TABLE 2: Subscripts and parameters of the problem formulation.

Notations	Definition
S	Number of stops
N	Number of trips
F	Number of time periods
$t_{s,n}$	Arrival time of the trip n on stop s
$f_s^b(t)$	Passenger boarding flow distribution function
$f_s^o(t)$	Passenger alighting flow distribution function
Q_s^b	Expectation of total passenger boarding flow
Q_s^o	Expectation of total passenger alighting flow
C_{veh}	Vehicle available cost
C_{ij}	Link cost of arc (i, j)
P_{ij}	Penalty of the infeasible time of arc (i, j)
T_f	Trip time in period f
h_{min}^f	Lower bound of the headway at period f
h_{max}^f	Upper bound of the headway at period f
C_{max}	Maximum capacity of the bus vehicles

TABLE 3: Sets and variables of the problem formulation.

Notations	Definition
P	Set of pull-out arcs
Q	Set of pull-in arcs
R	Set of the trip-link arcs
A	Set of all arcs, $A = PUQR$
D	Set of depots
T	Set of trips
h_f	Headway during the period f
x_{ij}	1 if arc (i, j) is selected; 0 otherwise

$$\text{s.t. } t_{s+1,n} - t_{s,n} = \frac{T_f}{S}, \quad s = 1, \dots, S, n = 1, \dots, N, \quad (2)$$

$$t_{1,n} - t_{1,n-1} = h_f, \quad n = 1, \dots, N, \quad (3)$$

$$h_{min}^f \leq h_f \leq h_{max}^f, \quad (4)$$

$$\sum_{s=1}^S \left(\int_{t_{s,n-1}}^{t_{s,n}} f_s^b(t) dt \cdot Q_s^b - \int_{t_{s,n-1}}^{t_{s,n}} f_s^o(t) dt \cdot Q_s^o \right) \leq C_{max}, \quad (5)$$

$$\sum_{i:(i,j) \in A} x_{ij} - \sum_{i:(i,j) \in A} x_{ji} = 0, \quad \forall j \in T, \quad (6)$$

$$\sum_{i:(d,j) \in A} x_{dj} - \sum_{i:(i,d) \in A} x_{id} = 0, \quad \forall d \in D, \quad (7)$$

$$\sum_{i:(i,j) \in A} x_{ij} = 1, \quad \forall j \in T, \quad (8)$$

$$x_{ij} \in \{0, 1\}, \quad \forall (i, j) \in A. \quad (9)$$

Formula (1) is the objective function, where Z_1 is the objective function to minimize the passenger travel cost, Z_2 is the objective function to minimize the vehicle scheduling cost, and Z_3 is the objective function to minimize the incompatible trip-link cost. Formula (2) is the assumption of vehicle running at constant speed in one period. Formula (3) is the assumption of vehicle leaving on schedule. Formula (4) is the departure interval constraint for decision variables h_f . Formula (5) is the passenger capacity constraint, which requires that the number of passengers on the bus does not exceed the capacity of the vehicle. Formulae (6) and (7) are the flow-conservation constraints. Formula (8) is the cover

constraint requiring that each trip must be covered exactly by one vehicle, and formula (9) is the 0–1 constraint for trip-link decision variables x_{ij} .

Considering the uncertainty, the passenger flow in function Z_1 is assigned as the time integral of passenger flow distribution to get more accurate estimation. In general, the passenger flow distribution is fitted with a certain function, such as normal distribution [20]. In objective function Z_1 , $\sum_{s=1}^S \sum_{n=1}^N (t_{s,n} - t_{s,n-1}) \cdot \int_{t_{s,n-1}}^{t_{s,n}} f_s^b(t) dt \cdot Q_s^b$ is the passenger waiting cost, α is a non-negative weight to adjust the penalty of crowding, and $F_c(n, s)$ is the passenger crowding function expressed as formula (10). This formula reflects that the passenger crowding cost increases with the passenger volume on the bus at different periods.

$$F_c(n, s) = \sum_{s=1}^S \left(\int_{t_{s,n-1}}^{t_{s,n}} f_s^b(t) dt \cdot Q_s^b - \int_{t_{s,n-1}}^{t_{s,n}} f_s^o(t) dt \cdot Q_s^o \right). \quad (10)$$

Objective function Z_2 is defined as the sum of the costs of the available vehicles and the links between trips. The costs of the available vehicles is assigned as $\sum_{(d,j) \in P} C_{veh} \cdot x_{dj}$, in which x_{dj} is defined as the pull-out trip links. The costs of the links between trips is expressed as $\sum_{(i,j) \in A} C_{ij} \cdot x_{ij}$.

Objective function Z_3 reflects the gap between the theoretical trip time and the realistic trip time, which would reduce the feasibility of the scheduling scheme. As suggested by Shen et al. [20], P_{ij} in function Z_3 denotes the penalty of the infeasible time of arc (i, j) and can be defined as

$$\begin{aligned} P_{ij} &= E(\text{IF}_{ij}^2) = \int_{T_j - \text{DH}_{ij}}^{+\infty} (t_i^e - T_j + \text{DH}_{ij})^2 f_i(t) dt \\ &= \int_{T_j - T_i - \text{DH}_{ij}}^{+\infty} (T_i + t - T_j + \text{DH}_{ij})^2 f_i(t) dt, \end{aligned} \quad (11)$$

where T_i and T_j denote the scheduled departure time of trip i and j , respectively; DH_{ij} denotes the deadhead time from arrival point of trip i to departure point of trip j ; $\text{IF}_{ij} = t_i^e - T_j + \text{DH}_{ij}$ is the infeasible time of an arc (i, j) ; and $f_i(t)$ is the probability density function about the duration of trip i .

3. The Multiobjective Hybrid Algorithm

Due to the NP-hard nature of the multiobjective optimization problem, the exact approach is difficult to solve the problem. Therefore, we develop a multiobjective hybrid algorithm to optimize the multiobjective integrated optimization problem. The algorithm is a combination of a self-adjust genetic algorithm (SGA), large neighborhood search (LNS) algorithm, and Pareto separation operator (PSO). The SGA is applied to construct a new headway, the LNS is employed to find the optimal trip-link chain under the headway, and the PSO is used to select the multiobjective optimal solution. The solution obtained by the hybrid algorithm is the scheduling scheme that consists of the headway and trip-link chain. Basic steps of the multiobjective hybrid algorithm are shown in Figure 7. Some definitions and the constructive approaches for the solution

structure, the SGA, the LNS, and the PSO are elaborated in Sections 3.1–3.4.

3.1. The Structure of Solution Set. As mentioned in Section 2, the solution of the multiobjective model consists of two parts: headway vector $T_g = (t_1, t_2, \dots, t_n)$ and trip chain set $D_g = \{X_1, X_2, \dots, X_M\}$. The vector element t_n in T_g represents the headway value of the n -th characteristic periods. The element X_M in D_g represents a feasible trip chain operated by vehicle M and can be expressed as $X_M = (x_i, \dots, x_j)$, in which x_i denotes the i -th trip in the trip set.

For example, suppose there are three trips and two characteristic periods in the model. The headway vector $T_g = (10, 8)$ refers to the headway being 10 minutes in period 1 and 8 minutes in period 2, respectively. The trip-link chain set $D_g \setminus = \{X_1\}$ refers to that vehicle 1 is to carry out all the trips based on T_g , and the trip-link chain set element $X_1 = (1, 3, 2)$ refers to the trips 1 and 3 and 3 and 2 that are linked as one chain.

3.2. The Self-Adjust GA for Headway Vector

3.2.1. Basic Frame of the Method $G(s)$. The GA is the most commonly used metaheuristic algorithm and performs well in operation research fields such as crew scheduling [21] and vehicle scheduling [22]. The GA search is based on a population of solutions and some operators used to perform the evolution in search process. The main operators in this method are selection, crossover, and mutation. The solution in GA is usually coded in finite length string, which is called chromosome. Starting from a population of chromosomes, all the chromosomes are evaluated using fitness function to rank the individuals. The crossover operator and mutation operator are responsible for the creation of new individuals.

In our self-adjust genetic algorithm $G(s)$, a chromosome represents a headway vector T_g . The new headway vector T_g at each iteration g is acquired from a pair of parent chromosomes, which are selected by the roulette wheel. Each gene in the chromosome T_g represents a headway value in one period. The process of the crossover operator and mutation operator is illustrated as Figure 8. The location of the chromosome in the crossover operator and mutation operator are selected randomly at each iteration. The offspring individual created in each iteration should satisfy the headway constraints in the model to ensure the solutions obtained from the algorithm are feasible in the real world. The basic steps of the $G(s)$ are shown in Table 4.

3.2.2. Self-Adjust Strategy of $G(s)$. Different from the normal GA approach in which population size is assigned as a fixed value on initialization, the population size in our method is adjusted with iteration g . That is, the population size is initially defined as three and is adjusted with the new feasible solution added. In general, the median value of the headway constraints is supposed to be the expected headway. Therefore, we define that there are three individuals in the initial population set P_g , one is the headway vector

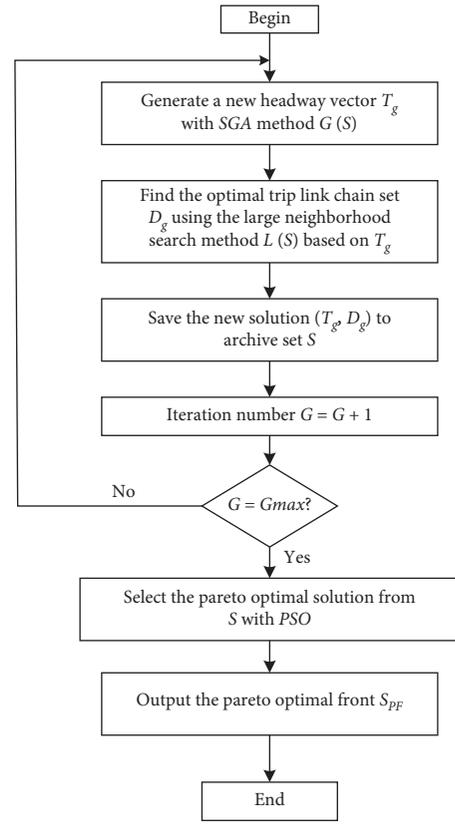
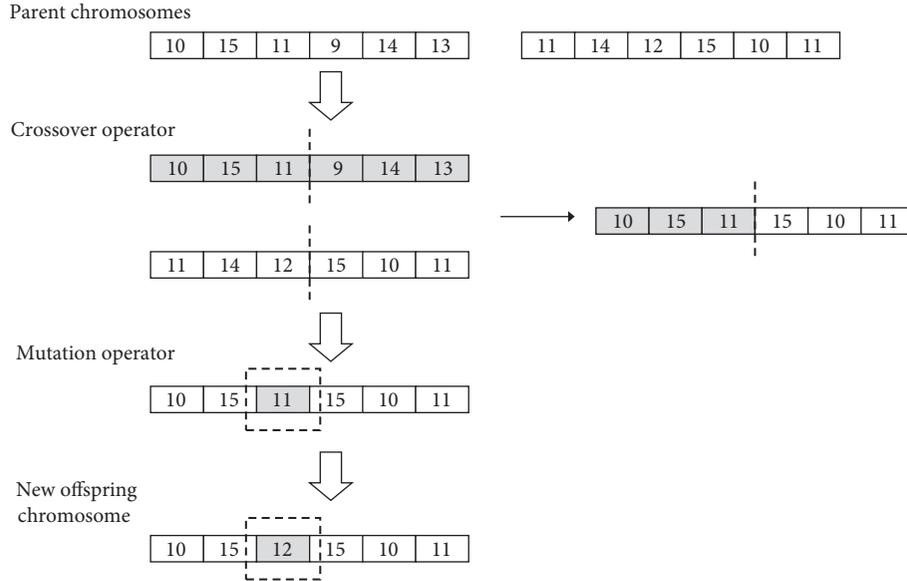


FIGURE 7: The framework of the multiobjective hybrid algorithm.

consisting of the median value of the headway constraints in each period, and others are the headway vectors plus one and reduce one, respectively, based on the median value headway vector. That is, the three initial chromosomes are $T_{s1} = (t_1, \dots, t_n)$, $T_{s2} = (t_1 + 1, \dots, t_n + 1)$, and $T_{s3} = (t_1 - 1, \dots, t_n - 1)$, where $t_f = (h_{\min}^f + h_{\max}^f)/2$, and h_{\min}^f and h_{\max}^f are the lower and upper bounds of the headway in period f . After each iteration, the new headway vector is added to the population set, and the population size is enlarged by one.

The parent chromosomes are selected using the roulette wheel at each iteration. Different from the normal roulette wheel method in which the selection probability of each chromosome in the population is calculated by fitness function, the selection probability of the three initial individuals is adjusted with the population size change. The selection probability of the initial individuals is calculated as $1/n$, where n is the number of individuals in the population set at each iteration. The fitness function value in the roulette wheel is the sum values of three objectives in formula (1).

3.3. The LNS Method for Trip-Link Chain. The LNS is originally proposed by Shaw [23] and has been successfully applied to the VSP and some extensions [24]. The basic idea of the LNS is to search in large neighborhoods, which may contain more potentially better solutions compared with small neighborhoods. The search neighborhood of the solution is defined implicitly by a destroy method and a

FIGURE 8: The crossover and mutation process of method $G(S)$.TABLE 4: Framework and steps of method $G(s)$.

Method	$G(s)$
1	Initialize population set $P_g = \{T_{s1}, T_{s2}, T_{s3}\}$
2	For $g = 1$ to G
3	Select a pair of parent chromosomes T_i, T_j from P_g using roulette wheel
4	Create a new chromosome T_g using crossover and mutation operators based on parent chromosomes
5	If T_g satisfy the headway constraints, go to step 6; otherwise, go to step 3
6	Add T_g to the population set as $P_{g+1} = \{P_g, T_g\}$
7	End for
8	Return the headway vector set $P_g = \{T_{s1}, T_{s2}, T_{s3}, T_1, \dots, T_n\}$

repair method. A destroy method disrupts part of the current solution, while a repair method rebuilds the destroyed solution. In the LNS, one or more destroy and repair methods would be employed. And in each iteration, the destroy and repair methods to be applied are selected according to their weights, which are defined by users before the iterations. The resultant solution is accepted based on an acceptance criterion, and the heuristic stops when the stop criterion is met.

The solution search of method $L(S)$ begins from an initial trip chain set D_g . This trip set is received by a greedy constructive heuristic method based on the timetable acquired from headway T_g . First, the trips are sorted in a nondecreasing order of their start times. Then, these trips are added to the solution during iteration. In each iteration, the selected trip is inserted to the end of the existing vehicle tasks that is compatible in time and leads to the minimum increase in the objective value of vehicle scheduling cost. If the trip is unfeasible to be added to any existing trip-link arc, it would be set as a new trip-link chain. The procedure stops until all the trips are served. The basic steps of the LNS method $L(S)$ are shown in Table 5.

The operators used in method $L(S)$ at each iteration are selected randomly. The iteration of method $L(s)$ is to search

trip link combination in solution neighborhood. The destroy operator is to remove some trips, and the repair operator is to insert these trips back to find new trip links. The definitions of the destroy operators and repair operators are described in the following sections.

3.3.1. Destroy Operators. A selection of destroy operators are implemented in our LNS, including random remove operator, short trip-link remove operator, and incompatibility trip-link remove operator. These removal operators are given different weights before iteration L and selected by the roulette wheel. The removal methods used in our LNS are defined as follows:

- (1) Random remove operator. It simply removes a number of trips randomly from the solution to diversify the search into different parts of the solution space.
- (2) Short trip-link remove operator. It would remove the shortest trip-link chain to reduce the vehicle scheduling costs.
- (3) Incompatibility trip-link remove operator. It would remove some maximal incompatibility penalty trip links from the solution to reduce the penalty costs.

TABLE 5: Framework and steps of method $L(S)$.

Method	$L(S)$
1	Initialize trip chain set $D_g = \{X_1, \dots, X_M\}$. Define the destroy operator set $\alpha^- = (\alpha_1^-, \dots, \alpha_r^-)$ and repair operator set $\beta^+ = (\beta_1^+, \dots, \beta_j^+)$
2	For $l = 1$ to L
3	Select a destroy method α_i^- and a repair method β_j^+ randomly
4	Remove q trips from D_g using α_i^- .
5	Insert the removed trips on D_g using β_j^+
7	End for
8	Return the final solution D_g

3.3.2. Repair Operators. A class of regret insertion heuristics is implemented as the repair operators in method $L(S)$. These operators insert one trip to the best position at a time. They differ in their ways of selecting the trip to be inserted [25]. These approaches could be defined as below:

Let N_r denote the set of trips to be inserted, and $\Delta f_{i,p}$ denotes the cost of inserting trip $i \in N_r$ to position p in the current partial solution. If trip i is inserted to position p , $\Delta f_{i,p} = c_{\text{pre}(p),i} + c_{i,\text{suc}(p)} - c_{\text{pre}(p),\text{suc}(p)}$, where $\text{pre}(p)$ and $\text{suc}(p)$ are the predecessor trip and successor trip corresponding to position p , respectively, and $c_{i,j}$ denotes the cost of trip-link arc (i, j) . If the insertion leads to an infeasible solution, we set $\Delta f_{i,p} = \infty$.

Let c_i^1 denote the insertion cost of trip i at its best position, $c_i^1 = \min_{p \in P} c_{i,p}$, where P is the set of all the positions, and c_i^k is the insertion cost at the k -th best position. In a regret- k heuristic, the trip $i^* = \text{argmax}_{i \in N_r} (\sum_{j=1}^{j=k} (c_i^j - c_i^1))$ is selected as the next trip to be inserted.

In our LNS, the regret-2, regret-3, and regret-4 heuristic methods are implemented and selected randomly with equal probability at each iteration. The trip insertion cost is calculated as the weighted sum of the vehicle scheduling cost and the penalty cost in formula (1).

3.4. The PSO to Simplify Computational Complexity

3.4.1. Pareto Dominance Relation. Multiobjective optimization deals with optimizing multiple objectives. As mentioned in Section 2, the integrated model formulates a problem with three objectives. Different from the optimization with one objective, a solution cannot be compared with another using relational operators in a multiobjective search space because objectives can be in conflict and require special considerations. This is due to the existence of more than one criterion for comparison [26]. Therefore, we need other operators to measure and find out how much a solution is better than another. The Pareto dominance is the most widely used operator to measure the solution of multiobjective model, and a series of multiobjective optimization algorithms are developed based on it [27]. The operator can be mathematically defined in a minimization optimization problem as follows:

$$\forall i \in \{1, 2, \dots, k\}: f_i(\vec{x}) \leq f_i(\vec{y}) \wedge \exists i \in \{1, 2, \dots, k\}: f_i(\vec{x}) < f_i(\vec{y}), \quad (12)$$

where $\vec{x} = (x_1, x_2, \dots, x_k)$ and $\vec{y} = (y_1, y_2, \dots, y_k)$. This equation shows that a solution (vector x) is better than another (vector y) if it has equal and at least one better value on all objectives and can be denoted as $x < y$. Suppose there are two objective functions f_1 and f_2 in a minimization optimization problem, the dominance relation could be presented from an example in Figure 9. This figure shows that the circles are better than squares because they provide a lower value in both objectives.

For the multiobjective optimization problem, there is a set of nondominated solutions. Consequently, the projection of Pareto optimal solutions in the objective space are stored in a set called Pareto optimal front. The optimization of multiobjective problem aims to find the Pareto optimal front of the problem.

3.4.2. Pareto Separation Operator. The Pareto dominance relation is applied in our multiobjective hybrid method to find the optimal operation scheme of the integrated model. Suppose there are m objectives and n acquired solutions in the multiobjective optimization problem, the number of comparison is $C_n^2 \cdot m$ in general, which would increase rapidly with the addition of the objectives and solutions. In order to reduce the number of comparison in our multiobjective algorithm and find the Pareto optimal front quickly, a separation solution comparison method which is called the Pareto separation operator is applied. The process of the PSO is presented as below.

It is easy to find that if one solution is nondominated in the multiobjective problem, it would be nondominated in at least one subproblem of the multiobjective optimization problem. That is, if one solution is nondominated in a two-objective subproblem of the multiobjective model, it would be a Pareto optimal solution. The relation between the multiobjective problem and the subproblem is illustrated in Figure 10, in which the point in the Pareto front is also the nondominated point in the projection. Therefore, in our comparison method, we only need to find the nondominated solutions in the two-objective subproblem of the multiobjective model, and the union of these solutions is the Pareto optimal solution set.

The comparison between two objectives is easier than the comparison between more objectives. Suppose there are n solutions in the two-objective subproblem, after we find the optimal solution S_m on the objective 1, we just need to

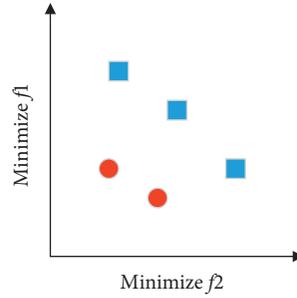


FIGURE 9: The Pareto dominance relation.

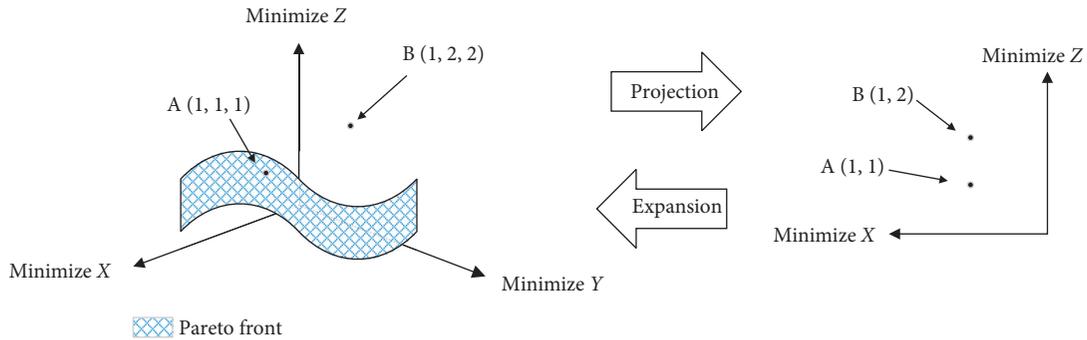


FIGURE 10: The relation between multiobjective problem and subproblem.

find the solution better than the solution S_m on the objective 2, and these solutions are the Pareto optimal solution of the two-objective subproblem. The number of comparison in our method would be no more than $C_m^2 \cdot 2(n - 1)$, where C_m^2 denotes the number of the two-objective subproblem. It is easy to prove that, if n is larger than $2(m - 1)$, $C_n^2 \cdot m$ would be larger than $C_m^2 \cdot 2(n - 1)$. In the most multiobjective problem, the number of solutions n is far larger than the number of objectives m . Hence, our method would effectively reduce the computational complexity. An example of the comparison method is shown in Figure 11.

4. Experiments and Results

To verify the effectiveness of the multiobjective optimization model, the experiments are carried out based on a case study on Route 3 of Jingmen Bus (JMB3) in China, which operates 128 trips and carries around 800 passengers per day. 23 buses are currently deployed on this route. The scheduled headway is about 5 to 15 minutes at most time throughout the day, with a maximum of 20 minutes in the afternoon off-peak period.

The experimental data preprocessing is presented in Section 4.1. This section proposes the approach based on history data to get the passenger flow distribution function, expectation of total passenger flow, and characteristic period division. The parameter setting of the multiobjective heuristic algorithm proposed in Section 3 is presented in Section 4.2. After that, the experimental results are presented in Section 4.3.

4.1. Passenger Flow Data and Trip Time Data Processing

4.1.1. Distribution Curve Fitting. The passenger flow and trip time would be changed over time. According to clustering method proposed by Fisher [28], the bus operating period could be divided to f periods with each period called a homogeneous running time (HRT) period [29]. In the same period, the passenger flow has the same distribution function, and the trips with departure time in same period have common time probability distribution. In this paper, the distribution is fitted with normal distribution. An example of the distribution curve fitting is shown as Figure 12, where the blue bar is the passenger flow and the red line is the fitting curve.

4.1.2. HRT Period Division. The HRT period is calculated on loss function. According to the method proposed by Fisher [28], the loss function values of different classification numbers are shown in Figure 13, in which the loss function values corresponding to the classification numbers of 5 and 6 are quite different, and there is not much difference from the loss function with the number of classifications being 6. Therefore, it is appropriate to divide the whole day into six characteristic periods in order to balance the efficiency and complexity of the method.

Based on the method proposed before, the HRT period division and trip time during each period are shown as Table 6.

4.2. Parameter Setting of the Multiobjective Hybrid Algorithm. In the study case experiments, the multiobjective algorithm executes 1000 iterations, that is, $g = 1000$ in the algorithm.

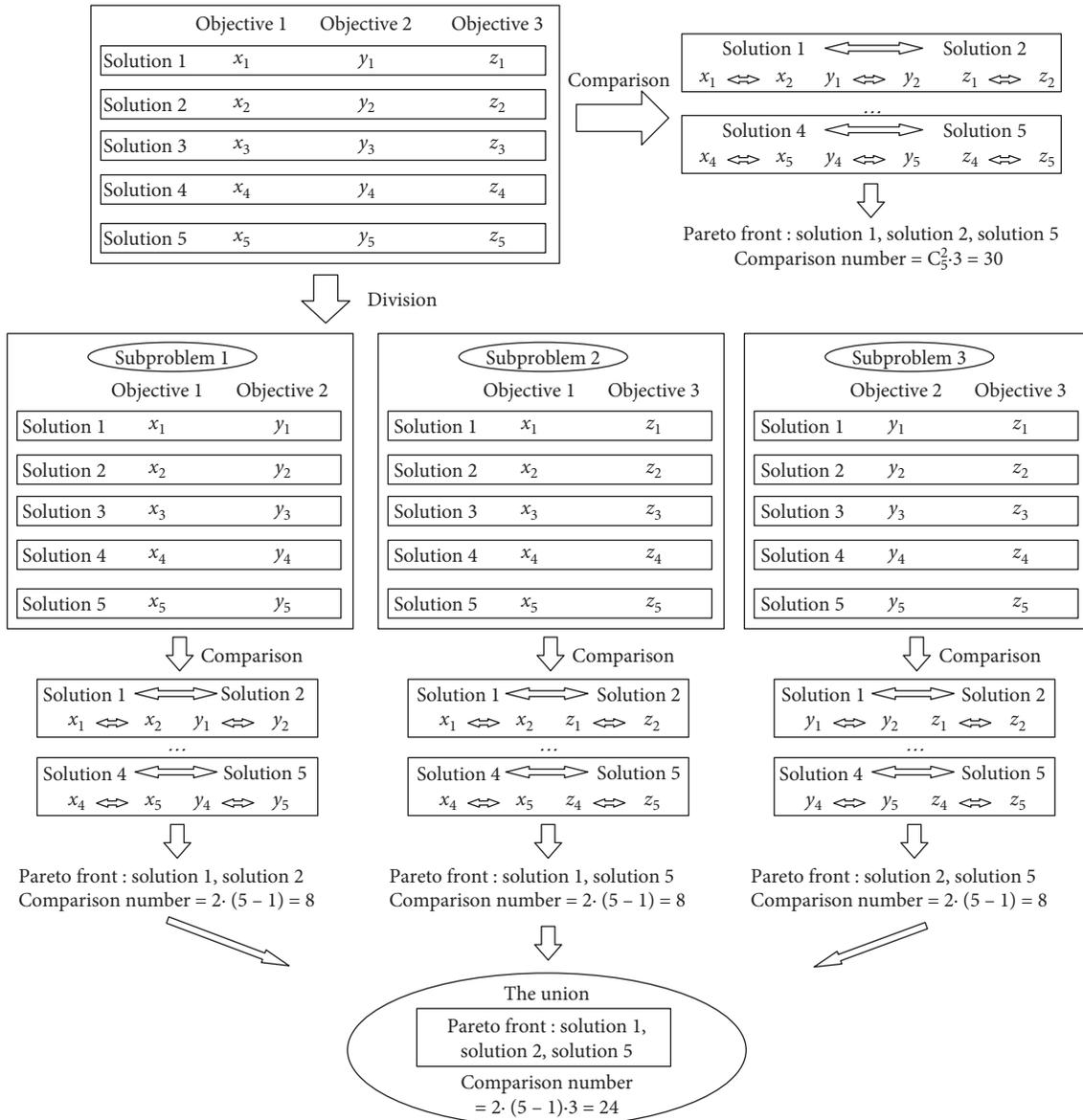


FIGURE 11: The example of the Pareto separation operator.

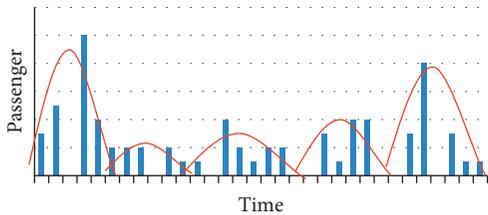


FIGURE 12: The normal distribution curve fitting of passenger flow distribution.

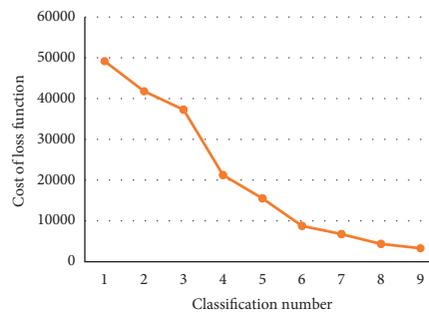


FIGURE 13: Loss function value of different classification numbers.

Meanwhile, the method $L(S)$ executes 1000 iterations. For the destroy operator 1 removes more trips and searches a larger neighborhood than other two destroy operators, the weight of destroy operator 1 is larger than others. The weights of three destroy operators in $L(S)$ are (0.5, 0.3, 0.2), and the random remove operator removes 10 trips randomly

at each iteration L . In the trip insertion, the weights of the vehicle scheduling costs and the penalty costs are assigned as 0.6 and 0.4 separately.

TABLE 6: Trip time during each period on Route 3 of Jingmen Bus.

Characteristic periods	Range of departure interval (min)	Expected value of trip time (min)
6:30-7:00	10-20	62
7:01-8:00	5-15	61
8:01-10:00	5-15	66
10:01-13:00	10-20	75
13:01-15:00	5-15	81
15:01-17:30	5-15	78

Up direction							Down direction					
Headway	15	10	10	15	10	10	15	10	10	15	10	10

Trip chain	Trips											
1	66	19	85	35	100	53	116					
2	60	6	65	110								
3	17	76	104	57	120							
4	88	51	114									
5	9	69	22	87	36	46	109					
6	1	3	64	39								
7	62	41	105	58								
8	78	28	94	47								
9	5	68	21	83	40	103	56					
10	13	72	91	112								
11	12	80	30	95	44	107						
12	67	20	79	29	93	42						
13	25	89	38	101	54	117						
14	4	63	48	111								
15	2	15	74	26	90	37	102	55	119			
16	18	77	27	33	97	59						
17	8	84	92	43	106							
18	7	82	32	96	49	113						
19	71	23	118									
20	61	14	73	81	31							
21	10	34	98	50								
22	11	70	24	86	45	108						
23	16	75	99	52	115							

FIGURE 14: An example of the integrated problem solution.

4.3. *Experimental Results.* Using the above data of the study case and the algorithm parameters, the MATLAB version R2017b is used to solve the integrated model. It runs on a personal computer (PC) with Windows 10, Intel Core i5, and 16 GB random access memory (RAM).

Figure 14 presents a solution of the integrated problem acquired through the multiobjective hybrid algorithm. Figure 15 illustrates the Pareto optimal solution set and initial solution set of the integrated model obtained by the multiobjective method. From Figure 15, we can see that our multiobjective hybrid algorithm is effective to find the Pareto optimal front. Our multiobjective method focuses on finding the solutions with lower vehicle scheduling cost and incompatibility penalty.

Figure 16 illustrates the comparison between the Pareto optimal front of the integrated model, the cost of recursive approach schedule, and the cost of the current implementation schedule on JMB3. The figure shows that the scheduling scheme of the integrated model is partially better than the recursive approach schedule and better than the current implementation schedule.

The fleet size and objective function value comparison among the Pareto optimal front, the recursive approach schedule, and the current schedule are shown in Table 7. Based on Euclidean distance, the closest point of the Pareto optimal solution to the recursive approach schedule and the current implementation schedule are shown in Figure 17.

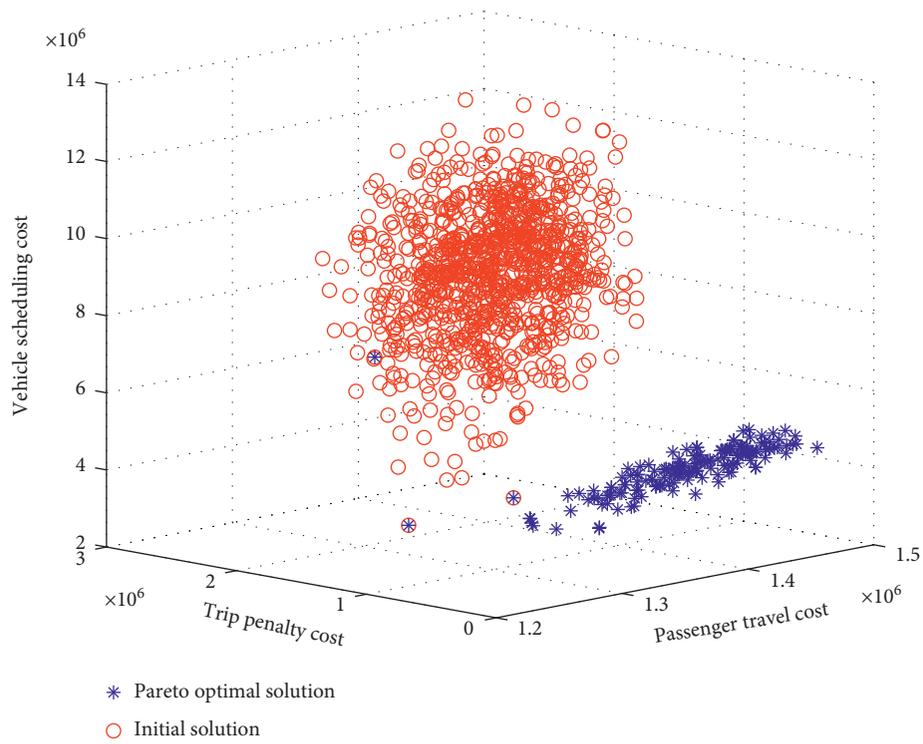


FIGURE 15: Pareto optimal solution set and initial solution set of the integrated model.

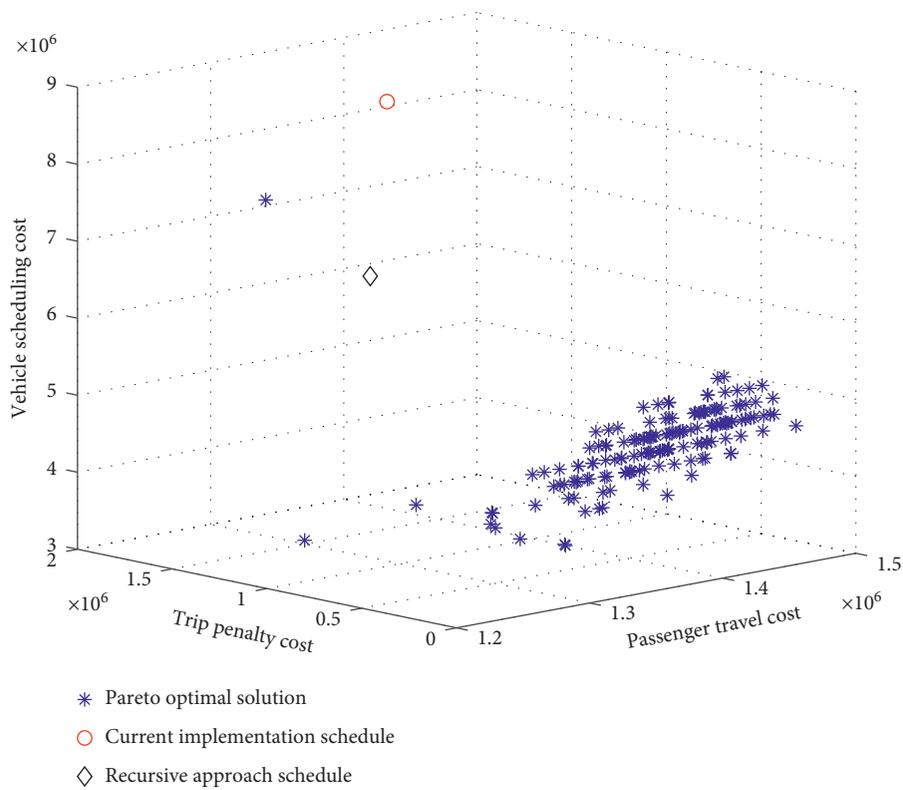


FIGURE 16: Solution costs of different schedules.

TABLE 7: Results of the different schedules.

	Fleet size	Passenger travel cost	Incompatible trip-link cost	Vehicle scheduling cost
Pareto optimal solution	20–23	$1.2 \times 10^6 \sim 1.4 \times 10^6$	$0 \sim 1.2 \times 10^6$	$3.6 \times 10^6 \sim 7.8 \times 10^6$
Recursive approach schedule	23	1.3×10^6	1.5×10^6	6.3×10^6
Current implementation schedule	23	1.3×10^6	1.2×10^6	8.7×10^6

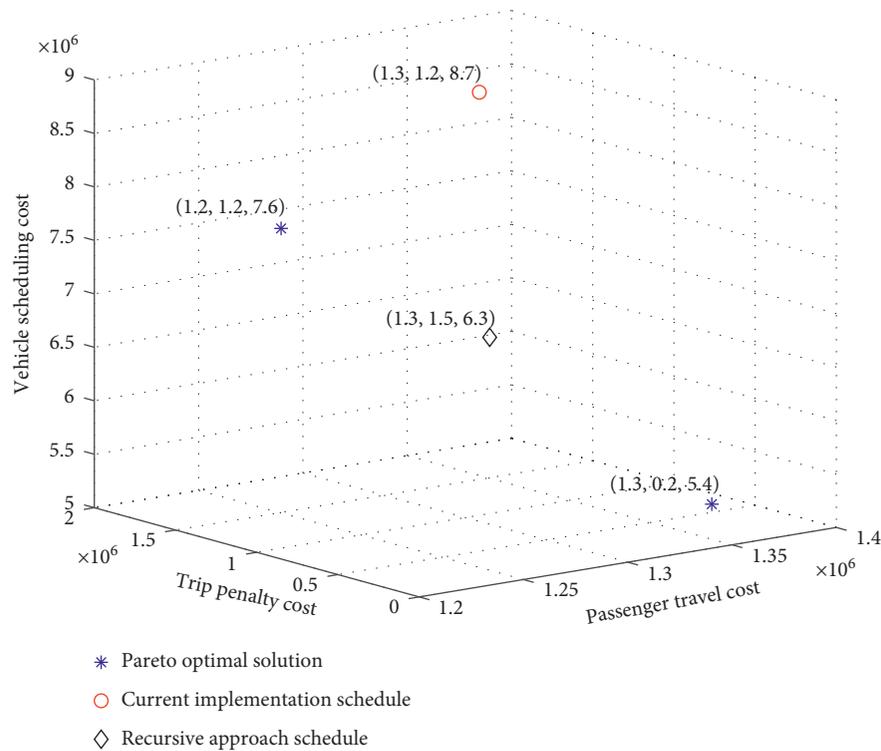


FIGURE 17: Closest Pareto optimal point to the different schedules.

5. Conclusion

This paper has proposed a multiobjective optimization approach for timetabling and vehicle scheduling with stochastic trip time and passenger flow. A multiobjective optimization model has been established comprehensively with the objectives of minimizing the passenger travel costs, vehicle scheduling costs, and incompatible trip-link costs while satisfying the passenger and bus corporation demands and improving the scheduling robustness by considering the uncertainty. In this paper, the objective of minimizing the passenger travel costs is different from the previous research projects, which optimize the timetabling based on headway smoothing rule by considering the passenger waiting cost and comfort cost. And the objective of minimizing the incompatible trip-link costs is to penalize the incompatible trip-link chain attributed to the stochastic trip time and enhance the robustness of the vehicle schedule. Meanwhile, the probability distribution of the stochastic trip time and passenger flow following are obtained by the measurement history data in advance. After the model establishment, a multiobjective hybrid algorithm, which is combined with

self-adjust GA, LNS, and Pareto separation operator, is proposed to solve the problem. A case study on real instance of Route 3 of Jingmen Bus (JMB3) is reported.

The experimental results on the problem instance derived from the JMB3 case show that our multiobjective optimization approach can produce the scheduling schemes that perform better than the current implementation schedule at one or more objectives. Compared with the recursive approach, the multiobjective hybrid method proposed in this paper can generate the scheduling scheme that balances the different costs.

Since the scheduling schemes generated by the multiobjective optimization method are a series of feasible schemes, our proposed approach in this paper would contribute to the transit network planning. Although the method in this paper performs well, there is still room for further improvement. For example, the model does not consider the stochastic deadhead time, and the probability distribution model of passenger flow and trip time is derived from limited data; hence, the probability distribution model is simple. These problems will require more work in the future.

Data Availability

All the data included in this study are available upon request by contact with the corresponding author (yindong@hust.edu.cn).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was supported by the National Nature Science Foundation of China (Grant no. 71571076) and by National Key R & D Program for the 13 Five-Year Plan of China (2018YFF0300301 in 2018YFF0300300).

References

- [1] K. P. Li, H. F. Huang, and P. Schonfeld, "Metro timetabling for time-varying passenger demand and congestion at stations," *Journal of Advanced Transportation*, vol. 2018, Article ID 3690603, 26 pages, 2018.
- [2] HY. Shang, YP. Liu, HJ. Huang, and R. Y. Guo, "Vehicle scheduling optimization considering the passenger waiting cost," *Journal of Advanced Transportation*, vol. 2019, Article ID 4212631, 13 pages, 2019.
- [3] J. Ma, A. A. Ceder, Y. Yang, T. Liu, and W. Guan, "A case study of Beijing bus crew scheduling: a variable neighborhood-based approach," *Journal of Advanced Transportation*, vol. 50, no. 4, pp. 434–445, 2016.
- [4] A. Ceder, "Public-transport vehicle scheduling with multi vehicle type," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 3, pp. 485–497, 2011.
- [5] M. Salicrú, C. Fleurent, and J. M. Armengol, "Timetable-based operation in urban transport: run-time optimisation and improvements in the operating process," *Transportation Research Part A: Policy and Practice*, vol. 45, no. 8, pp. 721–740, 2011.
- [6] M. Wei, B. Sun, and W. Jin, "A bi-level programming model for uncertain regional bus scheduling problems," *Journal of Transportation Systems Engineering and Information Technology*, vol. 13, no. 4, pp. 106–112, 2013.
- [7] O. J. Ibarra-Rojas, F. Delgado, R. Giesen, and J. C. Muñoz, "Planning, operation, and control of bus transport systems: a literature review," *Transportation Research Part B: Methodological*, vol. 77, pp. 38–75, 2015.
- [8] S. Bunte and N. Kliewer, "An overview on vehicle scheduling models," *Public Transport*, vol. 1, no. 4, pp. 299–317, 2009.
- [9] A. Ceder, "Efficient timetabling and vehicle scheduling for public transport," in *Lecture Notes in Economics and Mathematical Systems*, S. Vob and J. R. Daduna, Eds., vol. 505, pp. 193–217, 2001.
- [10] A. Ceder, "Optimal multi-vehicle type transit timetabling and vehicle scheduling," *Procedia - Social and Behavioral Sciences*, vol. 20, pp. 19–30, 2011.
- [11] V. Schmid and J. F. Ehmke, "Integrated timetabling and vehicle scheduling with balanced departure times," *OR Spectrum*, vol. 37, no. 4, pp. 903–928, 2015.
- [12] N. Kliewer, T. Mellouli, and L. Suhl, "A time-space network based exact optimization model for multi-depot bus scheduling," *European Journal of Operational Research*, vol. 175, no. 3, pp. 1616–1627, 2006.
- [13] H. L. Petersen, A. Larsen, O. B. G. Madsen, B. Petersen, and S. Ropke, "The simultaneous vehicle scheduling and passenger service problem," *Transportation Science*, vol. 47, no. 4, pp. 603–616, 2013.
- [14] S. Carosi, A. Frangioni, L. Galli, L. Girardi, and G. Vallese, "A matheuristic for integrated timetabling and vehicle scheduling," *Transportation Research Part B: Methodological*, vol. 127, pp. 99–124, 2019.
- [15] Y. Yue, J. Han, S. Wang, and X. Liu, "Integrated train timetabling and rolling stock scheduling model based on time-dependent demand for urban rail transit," *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 10, pp. 856–873, 2017.
- [16] O. J. Ibarra-Rojas, R. Giesen, and Y. A. Rios-Solis, "An integrated approach for timetabling and vehicle scheduling problems to analyze the trade-off between level of service and operating costs of transit networks," *Transportation Research Part B: Methodological*, vol. 70, pp. 35–46, 2014.
- [17] J. Teng and T. Chen, "integrated approach to vehicle scheduling and bus timetabling for an electric bus line," *Journal of Transportation Engineering, Part A: Systems*, vol. 146, no. 2, p. 306, 2020.
- [18] Y. Wang, A. D'Ariano, J. Yin, L. Meng, T. Tang, and B. Ning, "Passenger demand oriented train scheduling and rolling stock circulation planning for an urban rail transit line," *Transportation Research Part B: Methodological*, vol. 118, pp. 193–227, 2018.
- [19] Y. Shen, J. Xu, and J. Li, "A probabilistic model for vehicle scheduling based on stochastic trip times," *Transportation Research Part B: Methodological*, vol. 85, pp. 19–31, 2016.
- [20] A. Chen and C. Yang, "Stochastic transportation network design problem with spatial equity constraint," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 1882, no. 1, pp. 97–104, 2004.
- [21] Y. Shen, K. Peng, K. Chen, and J. Li, "Evolutionary crew scheduling with adaptive chromosomes," *Transportation Research Part B: Methodological*, vol. 56, no. 10, pp. 174–185, 2013.
- [22] C. Wang, H. Shi, and X. Zuo, "A multi-objective genetic algorithm based approach for dynamical bus vehicles scheduling under traffic congestion," *Swarm and Evolutionary Computation*, vol. 54, p. 667, 2020.
- [23] P. Shaw, "Using constraint programming and local search methods to solve vehicle routing problems," in *Proceedings of the Principles and Practice of Constraint Programming-CP98*, pp. 417–431, Pisa, Italy, October 1998.
- [24] M. Wen, E. Linde, S. Ropke, P. Mirchandani, and A. Larsen, "An adaptive large neighborhood search heuristic for the electric vehicle scheduling problem," *Computers & Operations Research*, vol. 76, no. 76, pp. 73–83, 2016.
- [25] J.-Y. Potvin and J.-M. Rousseau, "A parallel route building algorithm for the vehicle routing and scheduling problem with time windows," *European Journal of Operational Research*, vol. 66, no. 3, pp. 331–340, 1993.
- [26] S. Z. Mirjalili, S. Saremi, H. Faris, and I. Aljarah, "Grasshopper optimization algorithm for multi-objective optimization problems," *Applied Intelligence*, vol. 48, no. 4, pp. 805–820, 2018.
- [27] K. Deb, N. Padhye, and G. Neema, "Multi-objective evolutionary optimization-interplanetary trajectory optimization with swing-bys using evolutionary multi-objective optimization," *Lecture Notes in Computer Science*, vol. 4683, pp. 26–35, 2007.

- [28] W. D. Fisher, "On grouping for maximum homogeneity," *Journal of the American Statistical Association*, vol. 53, no. 284, pp. 789–798, 1958.
- [29] J. Xu and Y. Shen, "Setting scheduled trip time based on AVL data," *Journal of Transportation Systems Engineering and Information Technology*, vol. 12, no. 5, pp. 39–45, 2012.