Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 7494127, 11 pages
https://doi.org/10.1155/2022/7494127

Research Article

Optimization of Metro Trains Operation Plans Based on Passenger Flow Data Analysis

Jun Yang,[1,2] **Yinghao Tang**,[1] **Tan Ye**,[1] **Xiao Han**,[1] **and Mengjie Gong**[1]

[1]*Big Data and Internet of Things Research Center, China University of Mining and Technology-Beijing, Bejing 100083, China*
[2]*Key Laboratory of Intelligent Mining and Robotics, Ministry of Emergency Management, Bejing, China*

Correspondence should be addressed to Yinghao Tang; mytangyh@163.com

Received 19 June 2022; Accepted 6 September 2022; Published 20 September 2022

Academic Editor: Li Zhu

Metro intelligent system produces massive passenger flow and traffic data every day, among which route, station, and operation data are important for optimizing the train operation scheme. We collect passenger flow information of Shenzhen metro, analyze the passenger flow pattern and its distribution characteristics based on the data warehouse of the Hadoop platform, and optimize the train operation scheme in this paper. Using dynamic passenger flow data, an optimization model with train departure and dwell time as decision variables and passenger waiting time, passenger ride time, train full load ratio, and train operation balance as objectives is developed. An improved parallel genetic algorithm (GA) incorporating a simulated annealing algorithm (SAA) and an optimal individual retention strategy is used to find the optimal result. To verify the usefulness of the method, simulation experiments are conducted on the optimization model and method using the real passenger flow and train operation data of Shenzhen metro, and the simulation results are compared with the original plan.

1. Introduction

The metro system is characterized by large capacity, fast speed, high frequency, and punctuality. It has become one of the best schemes to alleviate urban traffic congestion [1]. Metro system produces a large number of passenger flow data [2] such as passenger origin-destination (OD) information and train operation data. Using big data to analyze passenger flow data can improve rail transit train transportation efficiency [3] and passenger satisfaction.

The intelligent construction of the metro is an important means to relieve the pressure of urban traffic, and train schedule optimization is one of the important ones [4]. In the metro system, passenger origin-destination (OD) information is very important. It can be used for the optimization of the metro train operation plan. The train operation plans are developed from historical traffic data. It determines the train's departure time at each station, its dwelling time at the station, and its arrival time at the station. It needs to meet some operational constraints such as train full load factor and travel time. Through the analysis of OD data and passenger flow data, we can optimize the train operation scheme to improve passenger satisfaction [5] and reduce the operation cost of the metro.

Lots of research have been performed on metro schedule optimization by many scholars. In terms of optimization models and optimization objectives. Wang et al. [6] proposed a mixed integer programming model based on time-varying demand, which minimizes the passenger waiting time and the number of passengers unable to transfer, using train capacity as a constraint. Zhang et al. [7] developed two nonlinear nonconvex programming models considering the variation of train frequency, train running time, and stopping time, and under the constraints of train operation and passengers getting on and getting off process, the train timetable with the minimum full passenger travel time is designed. Qu et al. [8] proposed a two-step optimization model to change the metro schedule, in which the train departure interval is used as a decision term to reduce the waiting time of people in the first-step model. In the second

step model, the total energy consumption of all trains is minimized by taking the train leave and arrival times at various stations as the decision terms. Wu et al. [9] proposed a multi-objective train schedule optimization method with the objectives of minimizing total energy consumption, average waiting time, and average maximum load deviation and demonstrated through a case study that the method can be used to reduce the total energy consumption, the maximum load deviation and the waiting time of passengers. Xie et al. [10] designed a synchronized metro schedule and stopping timetable optimization model for passengers and energy saving and demonstrated experimentally that it is very effective in reducing train energy consumption, running time, and delay probability. In terms of optimization methods, Wihartiko et al. [11] used an improved integer programming model of the genetic algorithm to solve the bus schedule problem in chromosome design, initial population recovery technique, chromosome reconstruction, and generation-specific chromosome extinction, respectively. Shang et al. [12] established a total passenger travel time model to minimize the total passenger travel time and proposed a spatial branching delimitation algorithm to solve the model. Wang et al. [13] proposed a linear weighted compromise algorithm and a heuristic algorithm to find the best solution for the bi-objective integer programming model with the train stopping time control. Guo et al. [14] proposed a mixed integer nonlinear programming model for generating optimal train schedules and maximizing interchange synchronization events, and then a hybrid optimization algorithm (PSO-SA) combining particle swarm optimization and simulated annealing is designed, and its superiority is proved by comparing with many algorithms. Tang et al. [15] combines the genetic algorithm and the simulated annealing algorithm to find the best result of an optimization model considering multiple constraints. Liu et al. [16] developed a mathematical model of it considering headway time distance and dwell time. Then an improved artificial bee colony algorithm is designed to solve this problem. Tang et al. [17] developed a bi-objective optimization model considering the minimization of full passenger waiting time and departure time and designed an improved nondominated ranking genetic algorithm (NSGA-II) for fast search of Pareto optimal solutions by using a specific coding scheme. Huang et al. [18] proposed a two-step model for matching metro passenger relationships and reducing the full waiting time of passengers, respectively, and designed a hybrid MCMC-GASA (Markov chain Monte Carlo genetic algorithm simulated annealing) approach to solve the problem.

A review of the literature shows that there has been extensive discussion and research by many experts in the area of the subway train schedule optimization problem, and in previous studies, it was common to assume a constant passenger flow model at a particular moment in time and then to optimize the train travel plan for that particular moment in time. The reality is that passenger flows vary dynamically with time distribution [19], and in previous train schedule optimization, the passenger flow distribution is often first assumed to be normal or some other distribution pattern. However, modeling passenger flow patterns in complex scenarios by such approximate estimation models is inaccurate, which may lead to the inapplicability of the optimization model to the normalized environment. With the rapid development of big data technology, big data analysis methods provide new methods and techniques for train schedule optimization in the metro. We collect historical passenger ticket card data from the metro AFC, clean the data through a Hadoop big data platform, and then calculate the passenger arrival rate at each station and the passenger disembarkation rate between stations distributed over time. A multi-objective train schedule optimization model that takes into account train movements and passenger demand is proposed. Then a parallel genetic algorithm (GA) incorporating a modified simulated annealing algorithm is designed and the optimal subindividual retention strategy is added to get the best result. We use the measured data of Shenzhen metro to evaluate the proposed model and a solution method, and the result shows that the method is effective and accurate.

Other parts of this article are as follows: in Section 2, we describe the methodology for AFC data acquisition and processing. In Section 3, we develop a multi-objective optimization model considering metro operations and passenger travel demand. In Section 4, we propose a parallel improved genetic algorithm incorporating simulated annealing algorithm to solve the multi-objective optimization function. Section 5 brings in the multi-objective optimization model based on real historical passenger flow data of the Shenzhen metro and solves the optimal solution. Finally, Section 6 gives the conclusion of this paper.

2. Data Acquisition and Processing

2.1. Description of Data. The raw data we capture is the ticket card information from the metro automatic fare collection (AFC) system. When a passenger through the gate to ride the subway, the passenger information is saved in the AFC system and a corresponding travel data set is generated. The data set includes start station address, start line, start station time, destination station address, destination line, and destination time. Shenzhen metro generates approximately 5.9 million records per day, each record containing more than 60 attributes. To facilitate data statistics in the future, the source data is cleaned and transformed, and only the fields we can use are retained, as shown in Table 1.

The start station ID (indicated by s_station) is the station number where the passenger enters the station. Start line ID (denoted by s_line) is the line where the passenger enters the station. Inbound time (denoted by s_time) is the time when the passenger entered the station. Destination station ID (denoted by d_station) is the station where the passenger left the station. The destination line ID (denoted by d_line) is the line on which the passenger exits the station. The exit time (represented by d_time) is the time the passenger left the metro station. Thus, a passenger's ride record can be expressed as

Table 1: Example of passenger travel data.

s_time	s_station	s_line	d_time	d_station	d_line
20-08-19 18 : 47 : 09	268030	268	20-08-19 19 : 10 : 05	268033	268
20-08-19 20 : 11 : 09	241019	241	20-08-19 20 : 31 : 29	241011	241

$$x = (s\_time, s\_station, s\_line, d\_time, d\_station, d\_line). \quad (1)$$

2.2. Data Processing. In recent years, big data analysis technology has been developing, and accordingly, big data platforms are becoming more and more advanced and perfect [20]. The core features of big data platforms are scalable distributed storage and efficient parallel data processing and computing capabilities. In this paper, we set up a multinode Hadoop platform and add the corresponding ecological components, such as Hive and HBase, and then complete data processing and model building in this big data platform.

To reduce data interference and computational effort, we take the raw data stored in HDFS for data cleaning and then use Hive to store the data. Calculations are performed using Hive to get the passenger arrival and disembarkation rates.

Calculate the number of passengers who take the metro at station j in the same line in the period t_1.

$$C^j_{\text{in\_station}} = \left(C | s\_time \in t_1, s\_line = lineid, j \in s\_station\right). \quad (2)$$

Count the number of passengers who leave stations j in the same line during period t_2.

$$C^j_{\text{out\_station}} = \left(C | d\_time \in t_2, d\_line = lineid, j \in d\_station\right). \quad (3)$$

Calculate the number of passengers who take the metro from station i and get off at station j in the period t_2.

$$C^j_i = \left(C | d\_time \in t_2, s\_line = lineid, d\_line = lineid, i \in s\_station, j \in d\_station\right). \quad (4)$$

The passengers' arrival rate at j stations can be calculated by dividing $C^j_{\text{in\_station}}$ by t_1.

$$\lambda^j_{t_1} = \frac{C^j_{\text{in\_Station}}}{t_1} \left(j \in \text{in\_station}\right). \quad (5)$$

The proportion of passengers leaving stations i can be calculated by dividing $C^j_{\text{in\_station}}$ by C^j_i.

$$\rho^j_i = \frac{C^j_i}{C^j_{\text{in\_station}}} \left(i \in \text{in\_station}, j \in \text{out\_station}\right). \quad (6)$$

3. Multi-Objective Optimization Model

To improve the operational efficiency of the metro, we develop a passenger flow data-driven dynamic optimization model of the metro train operation plan in this section based on the passenger flow and travel data preprocessed by the Hadoop platform described in the previous part. The optimization model considers both metro operation and passenger experience, including train operation stability and train loading efficiency, and passenger experience including passenger ride and waiting time and the number of passengers on the train. We use a metro line consisting of k metro stations and l trains [21] as the target of our study, specifying the starting station as station 1 and the ending station as station k. To quantify the various parameters to describe the mathematical model, to better match the actual situation of metro operations as well as to simplify the overall optimization model, the following assumptions are required in this paper to build the model in terms of both passengers and metro trains.

(1) Only one train can stop at the same station in the same direction of subway operation at the same time, and there will be no overtaking when parallel trains are running on the subway line.

(2) When the train enters the metro station, all passengers line up to get off and get on following the principle of "first off, then on, first to arrive, first to serve."

(3) The maximum capacity of each train is a fixed value. When the number of passengers waiting on the platform exceeds the capacity of the train, the remaining passengers need to continue to wait on the platform and wait for a train to arrive.

Assumption (1) is generally applicable to most urban transportation systems to ensure that trains operate in sequence. Assumption (2) is in line with the mainstream passenger queuing principle, and assumption (3) can improve the running stability of the train and the comfort of passengers.

3.1. Model of Train Operation. Describing the operation of a train is generally performed by train exit time, inter-station running time, entry time, and dwell time [22]. Given a train l and a subway station k, the travel interval between train l and its preceding train $l-1$ can be expressed as the difference between the exit times of the two trains at station k:

$$h_{(l,k)} = d_{(l,k)} - d_{(l-1,k)}, \quad (7)$$

where $d_{(l,k)}$ is the moment of departure of train l from station k and $d_{(l-1,k)}$ is the moment of departure of train $l-1$

from station k. $d_{(l,k)}$ can be represented by the moment $a_{(l,k)}$ when train l arrives at station k and the stop time $s_{(l,k)}$ at station k.

$$d_{(l,k)} = a_{(l,k)} + s_{(l,k)}. \tag{8}$$

The time $a_{(l,k)}$ at which the train arrives at station k can be described as the total of the train's departure time $d_{(l,k-1)}$ from the last station and traveling time $r_{(l,k-1)}$ between the two stations.

$$a_{(l,k)} = d_{(l,k-1)} + r_{(l,k-1)}. \tag{9}$$

The running time is usually a preset fixed value because the distance between stations is certain and the train runs in autopilot mode between the two stations.

The stopping time $s_{(l,k)}$ of train l at station k can be expressed by this equation:

$$s_{(l,k)} = s_{min} + a \cdot \frac{U_{(l,k)}}{2N_{\text{door}}} + b \cdot \frac{D_{(l,k)}}{N_{\text{door}}}, \tag{10}$$

where $s_{\min}$ is the minimum stopping time of the train, a and b are two parameters that denote the time required for a passenger to board and alight respectively, which can be obtained analytically, N_{door} is the number of trains opening their doors at stations, for the convenience of calculation, we assume that the passengers who are going to get on the train will consciously form two lines, and the passengers who are going to get off the train will form one line in the train, $U_{(l,k)}$ and $D_{(l,k)}$ denote the number of passengers getting on and getting off train l at station k, respectively, these two parameters can be estimated from the historical data.

In addition, to improve safe train operation, two adjacent trains need to satisfy the minimum headway time constraint, i.e., the difference between the arrival time of train l at station k and the departure time of the previous train $l-1$ from station k should be greater than a constant, which can be described as $d_{(l,k)} - d_{(l-1,k)} \geq H\text{min}$.

3.2. Model of Passenger Demand. The number of passengers in a train l when the train leaves the station k is $P_{(l,k)}$. It can be represented by the number of passengers $P_{(l,k-1)}$ in train l when it leaves station $k-1$, the number of passengers $D_{(l,k)}$ who get off from station k and the number of passengers $U_{(l,k)}$ who get on board at station k:

$$P_{(l,k)} = P_{(l,k-1)} - D_{(l,k)} + U_{(l,k)}. \tag{11}$$

There is a maximum amount of passengers that a train can carry when it is running. As a result, passengers may become stranded at stations during peak traffic. The number of passengers boarding the train at the station k is $U_{(l,k)}$. It can be expressed by the number of passengers $P_{(l,k)}^{\text{remain}}$ remaining in the train at station k and the number of passengers $W_{(l,k)}^{\text{wait}}$ waiting at station k:

$$U_{(l,k)} = \min\left(P_{(l,k)}^{\text{remain}}, W_{(l,k)}^{\text{wait}}\right), \tag{12}$$

where the number of remaining passengers in train l at station k is $P_{(l,k)}^{\text{remain}}$. It can be represented by the maximum number of passengers on board as $Q_{(l,\max)}$, the number of passengers on board as $P_{(l,k-1)}$, and the number of passengers off the train as $D_{(l,k)}$:

$$P_{(l,k)}^{\text{remain}} = Q_{(l,\max)} - \left(P_{(l,k-1)} - D_{(l,k)}\right). \tag{13}$$

The number of passengers waiting for train l at station k is $W_{(l,k)}^{\text{wait}}$. It can be expressed by the number of passengers $W_{(l-1,k)}^{\text{remain}}$ stranded at station k by the previous train $l-1$ and the number of passengers $\lambda_k(d_{(l,k)} - d_{(l-1,k)})$ arriving in the travel interval between adjacent train l and train $l-1$, where λ_k is the passenger arrival rate in the interval between two adjacent trains $(d_{(l,k)} - d_{(l-1,k)})$ [23].

$$W_{(l,k)}^{\text{wait}} = W_{(l-1,k)}^{\text{remain}} + \lambda_k\left(d_{(l,k)} - d_{(l-1,k)}\right). \tag{14}$$

The number of passengers $W_{(l,k)}^{\text{remain}}$ stranded by train l at station k can be described as

$$W_{(l,k)}^{\text{remain}} = W_{(l,k)}^{\text{wait}} - U_{(l,k)}. \tag{15}$$

The number of passengers on train l who get off at station k is $D_{(l,k)}$. It can be represented by the number of passengers who boarded at the previous stations as $\sum_{i=1}^{k-1} U_{(l,i)}$, and the passenger boarding and alighting ratio O-D matrix as $E_{(i,k)}$:

$$D_{(l,k)} = \sum_{i=1}^{k-1} U_{(l,k)} E_{(i,k)}. \tag{16}$$

Big data analysis techniques can be used to statistically analyze historical passenger flow data to determine the proportion of passengers boarding and disembarking at each stop.

3.3. Multi-Objective Optimization Function. The optimization of train schedules based on dynamic and uneven passenger flows mainly includes train operation optimization and passenger satisfaction optimization. The train operation optimization mainly includes reducing the deviation of the actual train capacity from the desired capacity and ensuring the balance of train operation. Passenger satisfaction optimization consists of reducing the waiting time in the station and the travel time between stations.

The waiting time J_1 of passengers at the platform is a sum of the waiting time of passengers who are stranded after the departure of the previous train and the waiting time of new arrivals in the interval between the operation of two trains. It can be expressed as

$$J_1 = \sum_{l=2}^{N} \sum_{k=1}^{M} \left(\frac{1}{2}\lambda_k \left|d_{(l,k)} - d_{(l-1,k)}\right| + W_{(l-1,k)}^{\text{remain}} \left|d_{(l,k)} - d_{(l-1,k)}\right|\right). \tag{17}$$

Passenger travel time is the sum of the time passengers who are on board when the train is running and the time passengers who wait on board when the train stops at each station and can be expressed as

$$J_2 = \sum_{l=1}^{N} \sum_{k=1}^{M} \left(\left(P_{(l,k)} - D_{(l,k+1)}\right) s_{(l,k)} + P_{(l,k)}\, r_{(l,k)}\right). \tag{18}$$

The train running balance J_3 can be expressed as the difference between the stopping times of two adjacent trains running between stations at each station, and can be expressed as

$$J_3 = \sum_{l=2}^{N}\sum_{k=1}^{M}\left(\left|r_{(l,k)} - r_{(l-1,k)}\right| + \left|s_{(l,k)} - s_{(l-1,k)}\right|\right). \tag{19}$$

The difference J_4 between the actual capacity of the train and the desired capacity of the train can be expressed as follows:

$$J_4 = \sum_{l=2}^{N}\sum_{k=1}^{M}\left|P_{(l,k)} - P_{\max}\right|. \tag{20}$$

Considering the above elements to be optimized, the multi-objective optimization function can be described as

$$\begin{aligned} &\min J = a \cdot J_1 + b \cdot J_2 + c \cdot J_3 + d \cdot J_4, \\ &s.t. \begin{cases} d_{(l,k)} - d_{l-1,k} \geq d_{(\min,k)}\big), \\ s_{(l,k)} \leqslant s_{\max}, \end{cases} \end{aligned} \tag{21}$$

where a, b, c, d denote the weights of each objective, which are set differently according to different optimization needs. It is vital to increase the values of a and b suitably during peak passenger periods in order to carry passengers rapidly and decrease waiting and journey times. The stability of train operation should be improved and the operating cost should be decreased during the low-peak time of passenger flow, thus the values of c and d need to be suitably increased. The weights can be set in a balanced manner, taking into account the stability of train operation and the length of time passengers must wait, during the stable period of passenger flow. In conclusion, when choosing the weights for each optimization target, it is important to take into account both the passenger flow and the optimization requirements. The best weights should be chosen after conducting numerous tests.

4. Solution Method

To find the best solution for the multi-objective optimization model proposed in the previous section, we designed an improved parallelized genetic algorithm and completed the algorithm implementation in Hadoop big data platform.

4.1. Improved Genetic Algorithm. Genetic algorithm is a computing model that models natural selection and biological evolution, and it is a way of searching for optimal solutions by simulating the natural evolutionary process. GA provides a number of benefits, including the capacity to handle continuous and discrete variables, the adaptability of constraint definition, the capacity to handle huge search spaces, and the capacity to provide numerous optimal or good solutions [24]. The simulated annealing algorithm is derived from the solid annealing principle and has shown to be quite successful in locating the global optimum for a variety of NP-hard combinatorial problems [25]. Starting from a certain initial temperature, the probabilistic abrupt change property of SA can help the objective function to obtain the global optimal solution in the desired time as the temperature decreases [26]. Given the benefits of these two methods, Gandomkar et al. [27] presented a hybrid algorithm that combines GA and SAA to optimize the distributed generation resource allocation problem.

The advantage of the genetic algorithm is that it can quickly search out the whole solution in the solution space, excellent global search ability, overcoming the fast descent trap problem of other algorithms; suitable for distributed computing, natural parallelism speeds up the convergence speed. Relatively, genetic algorithm local search ability is insufficient, a simple genetic algorithm is time-consuming and less efficient for search in late evolution. SAA has a relatively powerful local search ability [28], but it cannot make the optimization search process the most promising area. Therefore, we improved the genetic algorithm and designed an adaptive genetic algorithm incorporating a simulated annealing algorithm with an optimal individual replacement strategy as follows:

(a) Encoding: The code consists of the train's departure moment at the origin station and the stopping time at each station, using a real number code whose values are generated within the departure interval and stopping constraints. An individual in the initial population can be represented as $(h_1, h_2, \ldots, h_l, t_{11}, t_{12}, \ldots t_{1k}, t_{2k} \ldots, t_{l1}, t_{l2} \ldots, t_{lk})$, where l is the number of trains, k is the number of stations, h_i denotes the interval between the departure of the train i and the preceding train from the first station, and t_{ij} denotes the stopping time of the train i at the station j.

(b) Selection: The genetic algorithm uses the roulette wheel selection method, but the probabilistic selection is random, to retain the good individuals, we use the best individual replacement strategy, i.e., we replace the individuals with low fitness values with those with high fitness values, thus increasing the fitness of the offspring. The specific selection method is as follows:

(1) Find the individual x_b with the highest fitness by calculating the fitness of each individual in the current population, assuming that the number of individuals in the population is N.

(2) Calculate the probability $p(x_i)$ that an individual is selected and the cumulative probability $q(x_i)$.

$$\begin{aligned} p(x_i) &= \frac{\mathrm{Fit}(x_i)}{\sum_{i=1}^{N}\mathrm{Fit}(x_i)}, \\ q(x_i) &= \sum_{j=1}^{i} p\big(x_j\big). \end{aligned} \tag{22}$$

(3) Randomly generate N numbers between [0, 1] in the array m as the selection probability. If the cumulative probability $q(x_i)$ is greater than the

Input: < key, value >, where the key is individual in one population, and the value is fitness in one population.
Output: < key′, value′ >, where key′ is the best individual in the iterative process, and value′ is the best fitness value individual to key′.
Algorithm Procedure:

```
(1)  Identify the number of iterations as M.
(2)  Initiate integer i = 0.
(3)  While(i < M):
       Compute individual fitness;
       Compute individual cumulative probability;
       Select;
       Crossover;
       Mutation;
       i++;
       End while
(4)  Compute individual fitness;
(5)  Find the best chromosome and fitness;
(6)  Output the chromosome and fitness;
```

Algorithm 1: Mapper.

Input: < key, value>pair, where the key is the best individual in each population, and the value is the best fitness in each population.
Output: < key′, value′>pair, where key′ is the ideal individual in all populations, and value′ is the best fitness in all optimal individuals of the population.
Algorithm Procedure:

```
(1) Identify the number of population N;
(2)  For i = 1 to N:
       Find maximum fitness;
       End For
(3) Output the chromosome and fitness;
```

Algorithm 2: Reduce.

element $m[i]$ in the array, the individual x_i is selected, if it is less than $m[i]$, the next individual x_{i+1} is compared until an individual is selected.

(4) Repeat step 3 until N individuals are selected.

(5) Find the individual x_w with the lowest fitness by recalculating the fitness of the N freshly created individuals.

(6) Replace the worst individual x_w with the previously selected best individual x_b to form the next generation population.

(c) Crossover: Two individuals are selected for simulated binary crossover operation based on the set crossover probability, and then the child fitness value $\mathrm{Fit}(c)$ and the parent fitness value $\mathrm{Fit}(p)$ are calculated for the simulated annealing operation. Let T_0 denotes the initial temperature, α is a positive number less than 1 and generally takes values between 0.8 and 0.99 the temperature calculation formula is

$$T(n+1) = \alpha T(n). \tag{23}$$

The new state is accepted at annealing with a probability according to the Metropolis criterion.

$$P = \begin{cases} 1, \mathrm{Fit}(c) > \mathrm{Fit}(p), \\ \exp\left(\dfrac{\mathrm{Fit}(p) - \mathrm{Fit}(c)}{T}\right), \mathrm{Fit}(c) \le \mathrm{Fit}(p). \end{cases} \tag{24}$$

(d) Mutation: Regular polynomial variation encoded in real numbers for chromosomes that have completed the crossover operation according to a set probability of mutation.

4.2. Parallel Genetic Algorithm. Based on the improved genetic algorithm proposed earlier, we have proposed an improved parallel genetic algorithm. The specific algorithm is described as follows:

In Algorithm 1, Step (3) is the regular genetic operation, including selecting individuals with high fitness from the population and eliminating individuals with low fitness, crossing chromosomes with a certain probability, and mutating chromosomes with a certain probability. Step (4) is to compute the individual fitness after the iteration. Step (5) is to choose the optimal chromosome and fitness. Steps (6) is to output the

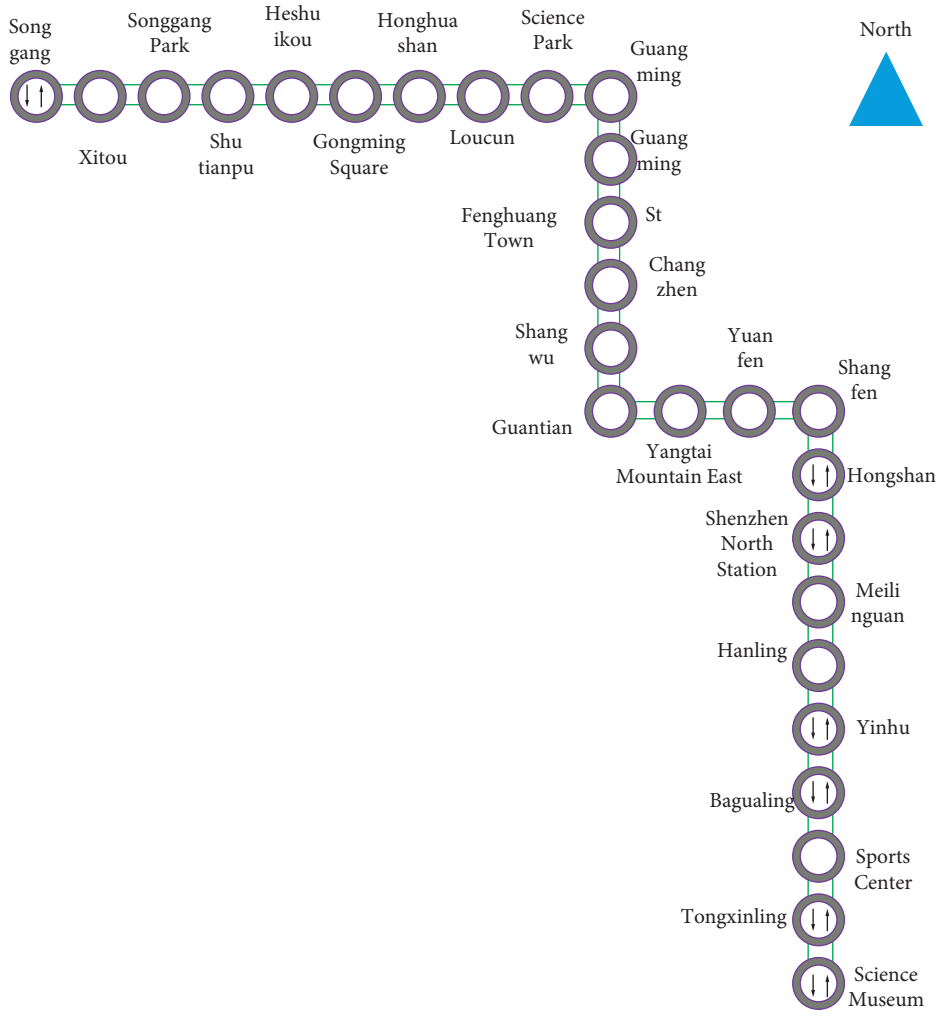

Figure 1: Station distribution of Shenzhen metro line 6.

Table 2: Stopping time of each station.

Station name	Stop Time(s)	Station name	Stop Time(s)
Songgang	40	Guantian	40
Xitou	40	Yangtai mountain east	40
Songgang park	40	Yuanfen	40
Shutianpu	40	Shangfen	40
Heshuikou	40	Hongshan	40
Gongming square	40	ShenzhenNorth station	45
Honghuashan	40	Meilinguan	40
Loucun	40	Hanling	40
Science park	40	Yinhu	40
Guangming	40	Bagualing	40
Guangming st	40	Sports center	40
Fenghuang town	45	Tongxinling	45
Changzhen	45	Science museum	40
Shangwu	40		

Table 3: Running time between stations.

Station section	Time(s)	Station section	Time(s)
Songgang-xitou	72	Shangwu-Guantian	104
Xitou-Songgang park	87	Guantian-Yangtai mountain east	246
Songgang park-shutianpu	136	Yangtai mountain east-Yuanfen	119
Shutianpu-Heshuikou	104	Yuanfen-Shangfen	132
Heshuikou-Gongming square	89	Shangfen-Hongshan	152
Gongming square-Honghuashan	120	Hongshan-ShenzhenNorth station	101
Honghuashan-Loucun	118	ShenzhenNorth station-meilinguan	191
Loucun-science park	86	Meilinguan-Hanling	160
Science park-Guangming	116	Hanling-Yinhu	146
Guangming-Guangming st	84	Yinhu-Bagualing	81
Guangming st-Fenghuang town	120	Bagualing-Sports center	68
Fenghuang town-changzhen	162	Sports center-tongxinling	70
Changzhen-Shangwu	238	Tongxinling-Science museum	82

Table 4: Detailed parameters of the train.

Type	Number of doors	Rated passenger capacity	Maximum passenger capacity
A	10	1860	2738

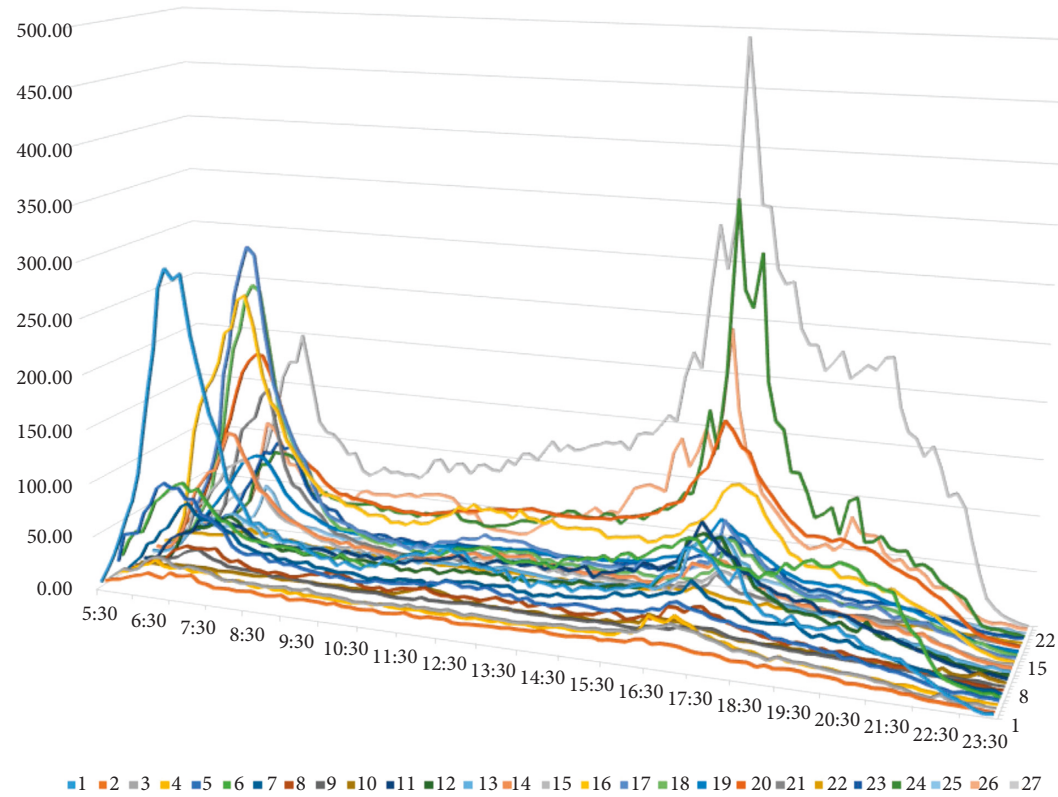

Figure 2: The passenger arrival rate of Shenzhen metro line 6.

intermediate <key, value> pair, and < key′, value′>pair. In Algorithm 2, Step (2) is to find the optimal chromosome and fitness in each population's optimal solution. Step (3) is to output the final chromosome <key, value> pairs and fitness value<key′, value′> pairs to the sequence file on HDFS.

5. Numerical Results

With the intention of verifying the performance of our designed optimization method in the multi-objective optimization model of the metro schedules, we collected the AFC data of Shenzhen metro line 6. The dataset

	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
1	0.01	0.05	0.03	0.08	0.15	0.04	0.02	0.01	0	0.06	0.03	0.02	0.06	0.04	0.06	0.03	0.04	0.07	0.11	0.01	0	0	0.01	0.01	0.01	0.02
2	0	0.05	0.03	0.08	0.16	0.04	0.03	0.01	0.01	0.07	0.03	0.02	0.04	0.03	0.11	0.04	0.02	0.07	0.11	0	0	0.01	0.01	0	0	0.01
3	0	0	0.03	0.09	0.15	0.05	0.03	0.01	0	0.06	0.03	0.03	0.05	0.04	0.1	0.04	0.03	0.07	0.12	0	0	0	0.01	0.01	0.01	0.03
4	0	0	0	0.05	0.12	0.1	0.05	0.02	0.01	0.09	0.03	0.03	0.04	0.04	0.08	0.04	0.03	0.08	0.12	0.01	0	0.01	0.01	0.01	0.01	0.01
5	0	0	0	0	0.14	0.11	0.06	0.02	0.01	0.1	0.06	0.03	0.06	0.03	0.07	0.03	0.03	0.08	0.13	0.01	0	0.01	0	0	0	0.02
6	0	0	0	0	0	0.09	0.06	0.03	0.01	0.2	0.06	0.06	0.05	0.03	0.07	0.02	0.03	0.09	0.14	0.01	0	0	0.01	0.01	0	0.02
7	0	0	0	0	0	0	0.04	0.01	0.01	0.21	0.1	0.05	0.08	0.03	0.08	0.03	0.05	0.09	0.15	0.01	0	0	0.01	0.01	0.01	0.03
8	0	0	0	0	0	0	0	0.01	0.01	0.2	0.07	0.07	0.06	0.05	0.12	0.04	0.04	0.1	0.16	0.02	0	0.01	0	0	0.01	0.02
9	0	0	0	0	0	0	0	0	0.04	0.19	0.08	0.05	0.07	0.04	0.15	0.02	0.02	0.11	0.17	0.01	0	0	0.01	0	0.01	0.02
10	0	0	0	0	0	0	0	0	0	0.21	0.06	0.07	0.06	0.05	0.13	0.04	0.05	0.12	0.18	0	0.01	0.01	0	0.01	0	0.01
11	0	0	0	0	0	0	0	0	0	0	0.1	0.07	0.1	0.08	0.12	0.04	0.06	0.11	0.19	0.02	0	0.01	0.02	0.01	0.02	0.05
12	0	0	0	0	0	0	0	0	0	0	0	0.03	0.12	0.05	0.14	0.07	0.15	0.11	0.19	0.02	0.01	0.01	0.02	0.02	0.02	0.04
13	0	0	0	0	0	0	0	0	0	0	0	0	0.08	0.07	0.17	0.08	0.11	0.14	0.22	0.03	0	0.01	0.02	0.01	0.02	0.03
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0.05	0.19	0.09	0.11	0.14	0.24	0.02	0.01	0.02	0.04	0.01	0.02	0.06
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.15	0.09	0.13	0.14	0.24	0.02	0.02	0.02	0.05	0.01	0.05	0.07
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0.15	0.11	0.23	0.07	0.02	0.03	0.06	0.02	0.04	0.17
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.21	0.05	0.16	0.07	0.03	0.02	0.12	0.02	0.08	0.24
18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.24	0.41	0.04	0.01	0.02	0.07	0.02	0.07	0.11
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.49	0.07	0.02	0.03	0.1	0.03	0.08	0.18
20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.15	0.05	0.07	0.21	0.07	0.17	0.28
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.05	0.04	0.21	0.09	0.21	0.4
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.06	0.2	0.14	0.35	0.25
23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.24	0.11	0.19	0.47
24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.2	0.28	0.52
25	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.22	0.78
26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Figure 3: Proportion of passengers getting off between stations.

Table 5: Initial information of the genetic algorithm.

Parameters	Value
Population size	50
Crossover rate	0.95
Variation rate	0.05
Genetic generation	500

Table 6: Departure interval of trains at the departure station.

Number	Headways	Number	Headways
Train 1	400	Train 10	395
Train 2	468	Train 11	376
Train 3	394	Train 12	366
Train 4	421	Train 13	409
Train 5	411	Train 14	384
Train 6	399	Train 15	395
Train 7	386	Train 16	435
Train 8	401	Train 17	435
Train 9	420		

contains a total of 15 million passenger trips and the data file size is over 25 GB. Shenzhen metro line 6 has a total of 27 stations, the distribution of which is shown in the map below (Figure 1).

The existing train schedules have fixed stopping times at each station as shown in the following Table 2.

Since the subway trains are in automatic mode, the train runs between two adjacent stations for a fixed period of time. This is shown in the following table (Table 3).

The train is a 6-part A-type train and the other information about the train are listed as follows. (Table 4).

The passenger arrival rate with time distribution is obtained using the historical passenger flow data statistics with the Hadoop big data platform for the study period. The following figure shows the distribution of passenger arrival rate at each station of Shenzhen metro line 6 over time in a day (Figure 2).

We decided to focus on two hours of the morning peak period to perform more precise schedule optimization research. In Figure 3, the statistical exit ratios between stations are displayed, where the final station is on the horizontal axis and the starting station is on the numerical axis. The data in the figure is 0, which means that few or no passengers get off from the station during the period.

A total of 17 trains are scheduled to depart during this period with a departure interval of 435 s. Using the departure interval of trains at the first station and stopping time at each station as the decision variables, the improved genetic algorithm introduced above is used to find the best result. The input information for setting up the genetic algorithm is listed below (Table 5).

The waiting time and travel time of passengers are the first optimization objectives, and the train operation balance is the secondary optimization objective. Therefore, the weights of the optimization function are set as $a = 0.4$, $b = 0.3$, $c = 0.2$, $d = 0.1$, respectively. The optimized train schedule does not increase the number of departures, and the departure interval of each train at the departure station is shown in the table below.Table 6.

The results of the comparison between the original train timetable and the optimized timetable are shown in Figure 4, where the horizontal axis is the arrival and departure time of trains at various stations and the vertical axis of each station of line 6 (Table 7).

The experimental results show that the optimized metro schedule reduces passenger waiting time by 21.42%, reduces passenger travel time by 22.56% and increases train full capacity by 2.65% compared to the existing schedule. It can be seen that the optimized metro timetable driven by

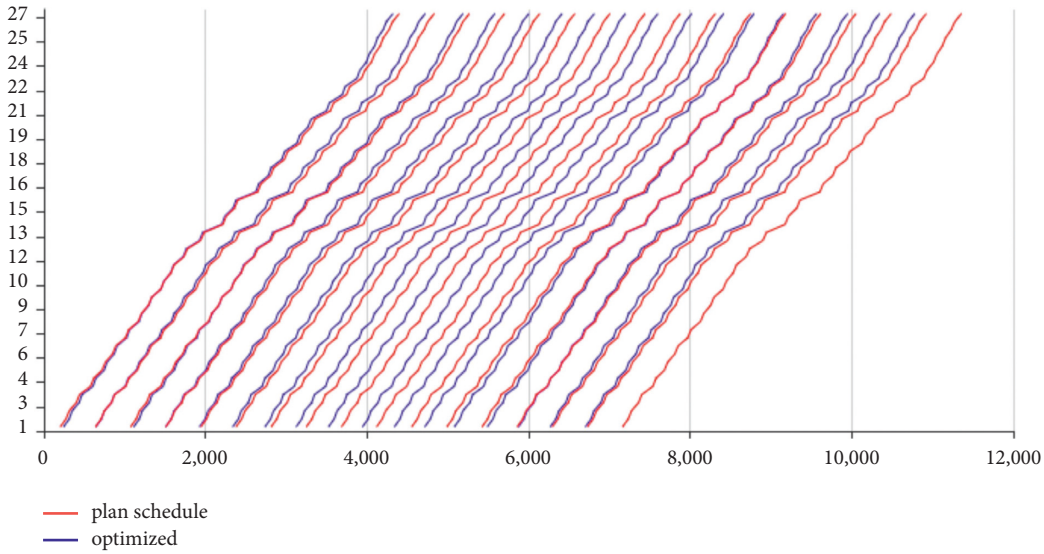

Figure 4: Comparison of original and optimized train schedules.

Table 7: Comparison of train schedule parameters before and after optimization.

Timetable	Original timetable	Optimized timetable
Passengers' total waiting time	32613	25625.1(−21.42%)
Total passengers' travel time	80235	62131.4(−22.56%)
Difference between actual train capacity and desired train capacity	321503	330032(+2.65%)

passenger flow data improves passenger satisfaction and train operation efficiency more than the existing planned schedule.

6. Conclusion

By analyzing and mining past passenger flow data, which the metro system creates in large quantities, it is possible to significantly increase operational efficiency and passenger pleasure. In this paper, we built a Hadoop big data platform to process and analyze the enormous historical passenger flow data of the Shenzhen metro, then we built a data warehouse to calculate the passenger inbound rate and the station-to-station disembarkation ratio of each station that changes at any time of the day through the Hive component. A multi-objective model considering both trains and passengers is proposed to optimize the train timetable. We have designed a parallel genetic algorithm incorporating simulated annealing algorithm improvements, using the best individual replacement strategy to retain the best individuals to get the best solution. Results of experiments using actual data from Shenzhen metro line 6 show that an improved train timetable can decrease passengers' waiting and transit times while also enhancing the balance of train operations and transportation effectiveness.

In future studies, we will further develop the proposed model with AFC data for multiple line interchanges. We will consider train operations for train turnarounds and turnbacks for the study, and another task to be performed is to analyze the passenger travel characteristics on holidays and weekends to optimize various nonworking day train schedules based on it.

Data Availability

Due to the nature of this research, participants of this study did not agree for their data to be shared publicly, so supporting data is not available.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was funded by the Beijing Municipal Natural Science Foundation (Grant No. L201015); the National Key R&D Program of China (Grant No. 2020YFC0833104); and The Green, Intelligent and Safe Mining of Coal Resources (Grant No. 52121003).

References

[1] E. B. Setyawan and D. Diah Damayanti, "Integrated railway timetable scheduling optimization model and rescheduling recovery optimization model: a systematic literature review," in *Proceedings of the 5th International Conference on*

Industrial Engineering and Applications (ICIEA), pp. 226–230, Singapore, April 2018.

[2] J. Feng, L. Liu, Q. Pei, and K. Li, "Min-max cost optimization for efficient hierarchical federated learning in wireless edge networks," *IEEE Transactions on Parallel and Distributed Systems*, p. 1, 2022.

[3] L. Zhu, H. Liang, H. Wang, B. Ning, and T. Tang, "Joint security and train control design in blockchain-empowered CBTC system," *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8119–8129, 2022.

[4] J. Yang, Y. Zheng, K. Yan et al., "SPDNet: a real-time passenger detection method based on attention mechanism in subway station scenes," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 7978644, 13 pages, 2021.

[5] L. Zhu, Y. Li, F. R. Yu, B. Ning, T. Tang, and X. Wang, "Cross-layer defense methods for jamming-resistant CBTC systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 11, pp. 7266–7278, 2021.

[6] Y. Wang, D. Li, and Z. Cao, "Integrated timetable synchronization optimization with capacity constraint under time-dependent demand for a rail transit network," *Computers & Industrial Engineering*, vol. 142, Article ID 106374, 2020.

[7] T. Zhang, D. Li, and Y. Qiao, "Comprehensive optimization of urban rail transit timetable by minimizing total travel times under time-dependent passenger demand and congested conditions," *Applied Mathematical Modelling*, vol. 58, pp. 421–446, 2018.

[8] Y. Qu, H. Wang, J. Wu, X. Yang, H. Yin, and L. Zhou, "Robust optimization of train timetable and energy efficiency in urban rail transit: a two-stage approach," *Computers & Industrial Engineering*, vol. 146, Article ID 106594, 2020.

[9] X. Wu, H. Dong, and C. K. Tse, "Multi-objective timetabling optimization for a two-way metro line under dynamic passenger demand," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 4853–4863, 2021.

[10] J. Xie, J. Zhang, K. Sun, S. Ni, and D. Chen, "Passenger and energy-saving oriented train timetable and stop plan synchronization optimization model," *Transportation Research Part D: Transport and Environment*, vol. 98, Article ID 102975, 2021.

[11] F. D. Wihartiko, A. Buono, and B. P. Silalahi, "Integer programming model for optimizing bus timetable using genetic algorithm," *IOP Conference Series: Materials Science and Engineering*, vol. 166, Article ID 012016, 2017.

[12] P. Shang, R. Li, and L. Yang, "Optimization of urban single-line metro timetable for total passenger travel time under dynamic passenger demand," *Procedia Engineering*, vol. 137, pp. 151–160, 2016.

[13] H. Wang, X. Yang, J. Wu, H. Sun, and Z. Gao, "Metro timetable optimisation for minimising carbon emission and passenger time: a bi-objective integer programming approach," *IET Intelligent Transport Systems*, vol. 12, no. 7, pp. 673–681, 2018.

[14] X. Guo, H. Sun, J. Wu, J. Jin, J. Zhou, and Z. Gao, "Multi-period-based timetable optimization for metro transit networks," *Transportation Research Part B: Methodological*, vol. 96, pp. 46–67, 2017.

[15] J. Tang, Y. Yang, and Y. Qi, "A hybrid algorithm for Urban transit schedule optimization," *Physica A: Statistical Mechanics and Its Applications*, vol. 512, pp. 745–755, 2018.

[16] H. Liu, M. Zhou, X. Guo, Z. Zhang, B. Ning, and T. Tang, "Timetable optimization for regenerative energy utilization in subway systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 9, pp. 3247–3257, 2019.

[17] J. Tang, Y. Yang, W. Hao, F. Liu, and Y. Wang, "A data-driven timetable optimization of urban bus line based on multi-objective genetic algorithm," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2417–2429, 2021.

[18] J. Huang, T. Zhang, and R. Wei, "Urban railway transit timetable optimisation based on passenger-and-trains matching – a case study of beijing metro line," *PRO*, vol. 33, no. 5, pp. 671–687, 2021.

[19] J. Yang, X. Dong, and S. Jin, "Metro passenger flow prediction model using attention-based neural network," *IEEE Access*, vol. 8, Article ID 30953, 2020.

[20] L. Zhu, F. R. Yu, Y. Wang, B. Ning, and T. Tang, "Big data analytics in intelligent transportation systems: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 1, pp. 383–398, Jan. 2019.

[21] Y. Li, L. Zhu, H. Wang, F. R. Yu, and S. Liu, "A cross-layer defense scheme for edge intelligence-enabled CBTC systems against MitM attacks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 4, pp. 2286–2298, 2021.

[22] Y. Wang, L. Zhu, Q. Lin, and L. Zhang, "Leveraging big data analytics for train schedule optimization in urban rail transit systems," in *Proceedings of the 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 1928–1932, Maui, HI, USA, November 2018.

[23] W. Xu, P. Zhao, and L. Ning, "A passenger-oriented model for train rescheduling on an urban rail transit line considering train capacity constraint," *Mathematical Problems in Engineering*, vol. 2017, Article ID 1010745, 9 pages, 2017.

[24] S. D. Dao, K. Abhary, and R. Marian, "A bibliometric analysis of Genetic Algorithms throughout the history," *Computers & Industrial Engineering*, vol. 110, pp. 395–403, 2017.

[25] Z. G. Wang, Y. S. Wong, and M. Rahman, "Development of a parallel optimization method based on genetic simulated annealing algorithm," *Parallel Computing*, vol. 31, no. 8–9, pp. 839–857, 2005.

[26] Z. Xin, L. Yu, L. Lianhui, Z. Jun, L. Yan, and H. Xiangdong, "A combination test suite generation method based on adaptive simulated annealing genetic algorithm for software product line testing," in *Proceedings of the 2nd Asia-Pacific Computer Science and Application Conference (CSAC 2017)*, pp. 641–648, 2017.

[27] M. Gandomkar, M. Vakilian, and M. Ehsan, "A combination of genetic algorithm and simulated annealing for optimal DG allocation in distribution networks," in *Proceedings of the Canadian Conference on Electrical and Computer Engineering*, pp. 645–648, Saskatoon, SK, Canada, May 2005.

[28] H. Liu, Z. Lin, Y. Xu, Y. Chen, and X. Pu, "Coverage uniformity with improved genetic simulated annealing algorithm for indoor Visible Light Communications," *Optics Communications*, vol. 439, pp. 156–163, 2019.