

Passenger Flow-Oriented Metro Operation without Timetables

Li He ¹, Lei Chen ^{1,*}, Jin Liu ², Clive Roberts ¹, Saijun Yu ³ and Xujie Feng ⁴

- ¹ Birmingham Centre for Railway Research and Education, University of Birmingham, Birmingham B15 2TT, UK; lxh553@student.bham.ac.uk (L.H.); c.roberts.20@bham.ac.uk (C.R.)
² Institute for Transport Studies, University of Leeds, Leeds LS2 9JT, UK; j.liu12@leeds.ac.uk
³ Hefei Urban Rail Transit Corporation, Hefei 230001, China; yusaidjun@hfgdjt.com
⁴ China Academy of Transportation Science, Beijing 100029, China; fengxujie@foxmail.com
* Correspondence: l.chen.3@bham.ac.uk

Abstract: Unpredictable fluctuant passenger flow usually exists in urban metro operations. In this situation, traditional predetermined metro timetables cannot always meet the variation of passenger flow, and thus the service quality of the metro system could be affected profoundly. In this paper, by introducing an innovative metro operation method without timetables, we develop a nonlinear integer programming model to continually optimise the train operation to deal with detected real-time passenger flow variations. We aim to minimise the total passenger waiting time in the research time horizon under the vehicle number constraint. A modified genetic algorithm integrated with a macroscopic metro simulator is adopted to solve the proposed model. A case study based on the Beijing Metro Line 19 is implemented to provide a quantitative result for evaluating the proposed passenger flow-oriented metro operation method without timetables. Compared to traditional timetable-based metro operation, the method could significantly improve the metro operation's flexibility and the quality of services.

Keywords: metro operation; genetic algorithm; passenger flow; optimisation



Citation: He, L.; Chen, L.; Liu, J.; Roberts, C.; Yu, S.; Feng, X. Passenger Flow-Oriented Metro Operation without Timetables. *Appl. Sci.* **2022**, *12*, 4999. <https://doi.org/10.3390/app12104999>

Academic Editors: Laith Abualigah, Absalom Ezugwu, Roberto A. Vazquez and Haruna Chiroma

Received: 14 April 2022

Accepted: 12 May 2022

Published: 15 May 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The metro system plays an essential role in urban public transportation. Because of its high speed, high capacity, and low energy consumption characteristics, the metro system has become the most popular type of urban transportation mode over the years, especially in large cities with many inhabitants, such as London, Tokyo, Beijing, Hong Kong, etc. Thus, plenty of effort is devoted to optimising the metro operation based on different demands.

From the macroscopic perspective, a typical metro operation problem is to make schedules for trains regarding systems' constraints. Unlike a mainline railway system operated and restricted to a predetermined timetable, periodic operation strategies are the most common solution for a metro system. However, this traditional method cannot always satisfy the variation of passenger flow. In general, the variation of passenger flow is catalogued into two main kinds: regular variation and irregular variation. Regular variation occurs when predictable changes occur, such as the passenger flow caused by rush hours, summer holidays, significant events, etc. These regular variations could be predicted based on historical experiences in the early stage, and operators have enough time to design specialised dynamic timetables. For example, the London Underground established a specialised timetable for massive passenger flow caused by the football match at Arsenal Football Club to ensure sufficient system capacity. Contrarily, irregular variations occur occasionally, and they are unpredictable; for example, due to a sudden change of weather, many tourists choose to take the metro instead of walking to a tourist attraction; these irregular variations will invalidate well-designed timetables, which can cause passenger oversaturation and impact the service quality. Usually, though the approximate passenger

flow distribution could be counted based on historical data, these real-time irregular variations will always exist, and predetermined dynamic timetables could not always satisfy these variations.

With the development of the automatic fare collection system and the face recognition technology, the real-time variations of passenger flow are easier to be detected. As a result, a new operation method could be proposed to adjust operation strategies to satisfy the real-time detected passenger flow variation. Though scheduling in a fixed timetable has the advantage of predictability for the passengers, it also makes the operation of the train easier for the drivers. Unlike a mainline railway system which limits the number of passengers boarding and provides a prepared timetable for passengers to plan their journey before purchasing the tickets, with its high frequency characteristic, most metro systems only publish the timetables for the first and last service (such as the Wuhan Metro system, the Beijing Subway system, etc.), most passengers do not plan their journey and board the first train arriving at the platform. Furthermore, based on a field study in the London Underground Bakerloo line operation department, because of the uncertainty in actual operation, metro drivers have to modify the operation strategies according to the metro operators' real-time instructions instead of following a prepared timetable. Thus, a real-time passenger flow-oriented operation method is more advantageous than a preprepared timetable for a metro system. This real-time operation method is essential to guarantee the quality of services, avoid safety issues caused by passengers' oversaturation in metro stations, and ensure the robust operation of the metro system. The associated service quality function can be formulated by a single or by multiple features of the considered metro system, including the overall travelling time of passengers, the waiting time of passengers, the transport capacity of the metro system in a time window, and the particular definition of passenger satisfaction, etc.

2. Literature Review

In recent years, rail operation problems have drawn tremendous attention from academia and industry [1–4].

A trendy area of rail operation research is train schedule. This research aims to propose different timetables according to different objectives. In general, three main goals are involved: avoiding train delays [5–8], reducing fuel consumption costs [9,10], and improving the quality of services [11]. For a passenger railway system, especially for an urban rail system, it is essential to design a passenger flow-oriented scheduling strategy to improve the service quality, system reliability, and passenger satisfaction [12].

The research on meeting the requirements of passenger flow-oriented scheduling can be divided into two categories. The first kind of research proposes periodic timetables [13–15], which means that trains are scheduled at a specific frequency and that timetables keep repeating in a time horizon. This periodic scheduling method is easy to formulate and optimise based on regular passenger flow variations and periodic events [4]. Thus, most metro systems in real-life apply this periodic operation strategy. However, because more passenger flow data can be collected and analysed, in recent years, instead of considering periodic timetables, more research has focused on scheduling to propose dynamic timetables based on dynamic origin–destination (OD) passenger demands [3,12,16,17]. For example, Yin et al. used the passenger OD matrix data at stations through a single day to minimise total passenger waiting time [18]. Halim et al. built an OD matrix to minimise the passenger waiting time and avoid high passenger loads [19].

The modelling method for scheduling is generally based on four main techniques: linear programming, integer programming (IP), nonlinear programming, and graph theory. For example, Petersen et al. used linear programming to develop a computer-assisted train dispatch system [20]. Ghoseiri et al. built a multi-objective train scheduling model using IP [21]. Li et al. proposed a green train scheduling model and fuzzy multi-objective optimisation algorithm based on nonlinear programming [22]. D'Ariano et al. used graph theory to solve the real-time perturbations for a rail system [23]. The solution methodologies can

be divided into three types: (1) commercial optimisation software, (2) heuristic algorithms (e.g., evolutionary algorithms [24]), and (3) simulation methods [25,26].

Another form of rail operation research is train rescheduling, which mainly focuses on adjusting the train operation into a regular timetable when operational incidents occur [27]. There are two main research directions: (1) Disturbance management deals with short incidents. In this situation, the dispatch strategy needs to be modified to reduce the influence on the timetable in the future [23,28]. (2) Disruption management deals with those events that cause a long delay. In this situation, it is not only the modification of the dispatch strategy that is needed—other measures, such as cancellation of trains, the addition of trains, and skipping stops, also need to be considered [29,30]. In general, rescheduling strategies can be divided into five categories [31]: (1) Holding, which is used to modify the headway between trains by holding the early-departing trains [32]. (2) Zone scheduling, where the whole line is divided into different zones, and all undeparted trains are divided into different groups, and a specific train only stops at all stations within a single zone. This scheduling method can deal with passengers piling up in a few specific stations [33]. (3) Short turning, a strategy that arranges several trains to only serve some zones with high demand defined by passenger flow speed and turn back [34,35]. (4) Deadheading, a strategy in which several trains are dispatched without stopping and collecting any passengers at the beginning of their trips to reduce the headways with other trains in front [36]. (5) Skipping strategy, used by most metro systems with high passenger demands, allows some late trains online to skip low-demand stations to recover the timetable.

Real-time rail operation research is also popular in the current decade. Some existing research has focused on fast optimisation for a specific operation in a particular circumstance to increase the stability of the system and avoid delay, such as the authors of [24,27,37] considering the optimisation of the trains' order in the junction area to minimise delay. The other real-time research is about rescheduling after an incident; this research focused on correcting the timetable to normal to minimise the impacts from disturbance or disruption that already occurs, such as [29,38,39]. As is apparent from the above review, most rail operation research is based on a predefined timetable, and most scheduling-related research is engaged to propose a robust timetable (dynamic or periodic) according to different requirements. Most rescheduling-related research focuses on reducing the impact on the timetable from incidents to recover trains' operation or to optimise a specific operation to increase system stability. The comparisons of representative metro operation research and this research are listed in Table 1 below.

Table 1. Comparisons of representative metro operation research.

Studies	Modelling Method	Methodologies	Proposal
Cordone et al. (2011)	Mixed-integer nonlinear programming	Heuristic algorithm	Periodic timetables
Yin et al. (2017)	Mixed-integer linear programming	Heuristic algorithm	Dynamic timetables
Ghoseiri et al. (2004)	Integer programming	Epsilon constrained method	Dynamic scheduling method
Li et al. (2013)	Nonlinear programming	Fuzzy multi-objective optimisation algorithm	Dynamic scheduling method
D'Ariano et al. (2007)	Graph model	Graph theory	Rescheduling after disturbance
Lei et al. (2012)	Mixed-integer linear programming	Heuristic algorithm	Real-time traffic management in junction areas
This research	Integer programming	Heuristic algorithm and simulation method	Real-time passenger flow monitoring based metro operation method

In this paper, the authors focus on proposing a new metro operation method to satisfy real-time detected passenger flow without timetables. As a result, the overall objective is to establish a systematic methodology to keep adjusting operation strategies for an urban metro line to replace predetermined timetables, which is applied to reduce the total waiting time for all passengers under the vehicle number constraint. The main innovations of this research are:

1. Proposal for a real-time passenger flow-monitoring-based urban metro operation method without timetables;
2. Formal formulation of the proposed metro operation method;
3. Development of a modified genetic algorithm to solve the problem.

This paper is organised as follows. The overall research motivations and background are introduced in Section 1; Section 2 gives a general introduction and review of related research. Section 3 formulates the real-time passenger flow-oriented operation problems for urban metro systems. A “real-time passenger-oriented operation method (RPOM)” based on nonlinear integer programming is proposed. To solve RPOM, Section 4 introduces an innovative algorithm, GA_RPOM, based on a modified generic genetic algorithm and a macroscopic metro system simulator. In Section 5, the proposed method is tested and evaluated based on the data from the Beijing Metro Line 19, and the results are compared with the other operation strategies with timetables in real life. Conclusions are presented in Section 6.

3. Mathematical Description

Two figures are applied to describe the problem. In Figure 1, there is a metro system; in general, trains will depart and stop at all the stations. We assume the passenger flow monitoring system, which can record passengers’ origins and destinations (ODs) based on their tickets, has been installed in all the stations. Thus, the average speed of entering passengers in a fixed period with different ODs could be counted. Passengers can only travel from a station labelled with a lower number to one marked with a higher number.

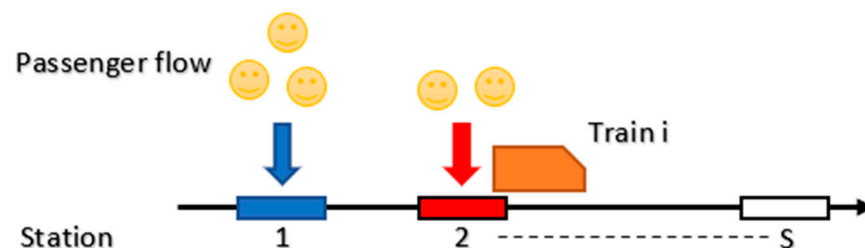


Figure 1. System description.

The process flow chart of RPOM is shown in Figure 2. The passenger flow entering speed will be kept up-to-date in every new detecting period, and the operation strategy will be optimised based on the objective function after the updating. As this process needs to be completed rapidly, the optimising method is essential.

Based on the above description, the considered problem should be formulated to a RPOM that constantly optimises the operation strategy based on the updated data. The following part focuses on the model’s details, including parameters, decision variables, constraint conditions, and objective functions. Several assumptions and explanations should be given first to make the scheduling process straightforward.

(A1) Discrete time, i.e., $1, 2, \dots, T$, will be used in this research;

(A2) Between stations, trains will run based on the predetermined running time, unless it is against safety constraints;

(A3) With the implementation of platform screen doors (PSD) and the deployment of station staffs, the impacts of passenger flow on the trains’ dwelling times have been significantly decreased in recent years. Thus, we assume the generated dwelling strategies can be followed strictly.

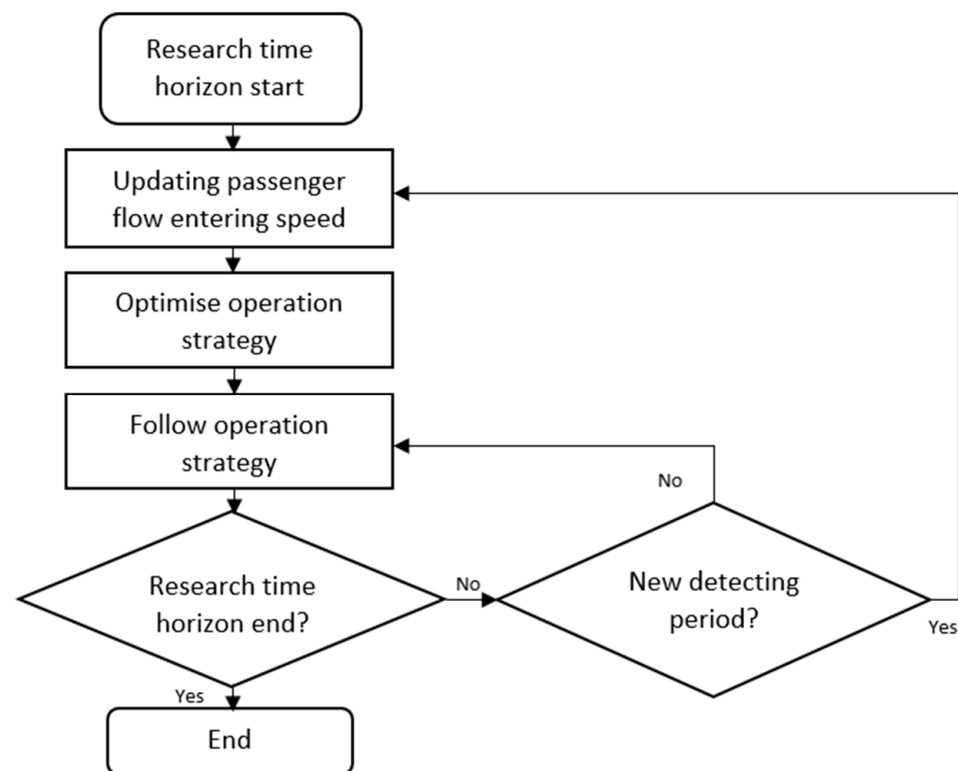


Figure 2. Real-time, passenger-oriented operation method (RPOM) process.

Based on the Field Study in the London Underground Bakerloo Line Operation Department, Modifying the Number of Trains in a Specific Operation Time Horizon Will Severely Affect the Economic Operation of the Metro System and Lead to More Operation Problems, such as the Staffs’ Redeployment. Thus, the Number of Trains Active in the Research Time Horizon Is Fixed for in this Research.

3.1. Decision Variables

The arrival and departure times have always been defined as two crucial decision variables for a traditional railway operation scheduling problem. However, with a set of optional departure intervals and dwells, arrival and departure times can be represented by the accumulation of departure intervals, dwell times, and running times. Thus, the decision variables involved in this problem are the choices of the intervals’ level and the dwell time level for each train at different stations, and binary codes could represent these choices. For clarity, the detailed definitions of decision variables are listed in Table 2.

Table 2. Decision Variables.

Notation	Definition
$ld_{i,s}$	Choice of dwelling time level for train i at station s , represented by binary codes.
$lh_{i,i-1}$	Choice of departure time interval level between train i and train $i - 1$ at station 0, represented by binary codes.

3.2. Notation and Parameters

Table 3 lists all the relevant notations and parameters in the problem.

Table 3. Notations and Parameters in this Problem.

Notation	Definition
\bar{S}	Set of stations in the metro system
\bar{I}	Set of train services
T	Set of times in the research time horizon
i	Index of train service numbers
s	Index of station numbers
t	Index of times
t_D	Detecting time
$r_{i,(s,s+1)}^{pre}$	Predetermined running time for train i to travel from station s to station $s + 1$
$r_{i,(s,s+1)}^{ope}$	Operational running time for train i to travel from station s to station $s + 1$
Ω_s^i	Set of optional dwelling times for train i at station s
$\omega_{l_{d_{i,s}}}$	Dwelling time for train i at station s , according to the choice of dwelling time level
H_i	Set of optional original time interval for train i
$h_{ll_{h_{i-1}}}$	Original time interval between train i and train $i - 1$ at station 0, according to choice of departure time level
$h_{i,i-1,s}$	Time interval between train i and train $i - 1$ at station $s, s > 0$
h_{min}	Minimum time interval between two trains
$a_{i,s}$	Arrival time of train i at station s
$d_{i,s}$	Departure time of train i at station s
$\lambda_s^s(t)$	Average passenger entering speed at station s to station s'
$p_{i,s}^{station}$	Passengers at station s waiting for train i
$p_{i,s}^{left}$	Passengers left by train i at station s
C_i^{max}	Maximum capacity of train i
$c_{i,s}$	Capacity of train i after passengers get off at station s
$p_{i,s}^{off}$	Number of passengers who want to get off at station s on train i
$p_{i,s}^{on}$	Number of passengers who can get on train i at station s
$p_{i,s}^{train}$	Number of passengers on train i when it is departing from station s

3.3. System Constraints

The dwelling time could be adjusted in operation, and, in doing so, the time interval between trains would be changed. Therefore, a lower bound for the time interval should be provided to guarantee safety in operation.

$$h_{i,i-1,s} \geq h_{min} \tag{1}$$

Meanwhile, the departure time interval between two trains in the first station should be included in the collection of optional intervals.

$$h_{ll_{h_{i-1}}} \in H_i \tag{2}$$

The dwelling time for each train should be chosen from the optional dwelling times set.

$$\omega_{l_{d_{i,s}}} \in \Omega_s^i \tag{3}$$

The number of passengers on a train after its departure from a station should be less than or equal to the train’s maximum capacity.

$$p_{i,s}^{train} \leq C_i^{max} \tag{4}$$

The arrival time of train i at the first station equals the departure time of train $i - 1$ plus the departure time interval according to level choice for train i .

$$a_{i,0} = d_{i-1,0} + h_{ll_{h_{i-1}}} \tag{5}$$

The operational running time between different stations is predetermined if the time interval between two trains is larger than the minimum interval; otherwise, it should be increased to guarantee the minimum interval.

$$r_{i,(s,s+1)}^{ope} = \begin{cases} r_{i,(s,s+1)}^{pre}, & r_{i,(s,s+1)}^{pre} + d_{i,s-1} - d_{i-1,s} \geq h_{min} \\ h_{min} - d_{i,s-1} + d_{i-1,s}, & r_{i,(s,s+1)}^{pre} + d_{i,s-1} - d_{i-1,s} < h_{min} \end{cases} \quad (6)$$

The arrival time $a_{i,s}$ of train i at station s equals the sum of the departure time at station $s - 1$ and the operational running time between these two stations.

$$a_{i,s} = d_{i,s-1} + r_{i,(s,s+1)}^{ope}, \quad 0 < s, s \in \bar{S} \quad (7)$$

The departure time $d_{i,s}$ of train i at station s is equal to the sum of the arrival time $a_{i,s}$ and the dwell time $\omega_{id_{i,s}}$.

$$d_{i,s} = a_{i,s} + \omega_{id_{i,s}} \quad (8)$$

The time interval between train i and train $i + 1$ at station s is equal to the arrival time $a_{i,s}$ minus the departure time $d_{i-1,s}$.

$$h_{i-1,i,s} = a_{i,s} - d_{i-1,s} \quad (9)$$

3.4. Passenger Constraints

The number of passengers at station s waiting for train i can be represented by the passengers who were left by train $i - 1$ and newly arrived passengers.

$$p_{i,s}^{station} = p_{i-1,s}^{left} + \sum_{d_{i-1,s+1}}^{d_{i,s}} \sum_{s+1}^{\bar{S}} \lambda_s^{s'}(t) \quad (10)$$

The remaining capacity of train i at station s immediately after passengers alight is given by:

$$c_{i,s} = C_i^{max} - (p_{i,s}^{train} - p_{i,s}^{off}) \quad (11)$$

The number of passengers who can get on train i at station s should be chosen from a comparison of the remaining train capacity and the number of passengers at the station.

$$p_{i,s}^{on} = \min(c_{i,s}, p_{i,s}^{station}) \quad (12)$$

If the remaining capacity is enough, all passengers can get on the train. Otherwise, some passengers will be left. The number of passengers left by train i at station s is given by:

$$p_{i,s}^{left} = \max(p_{i,s}^{station} - c_{i,s}, 0) \quad (13)$$

The number of passengers on train i when it is departing from station s is given by:

$$p_{i,s+1}^{train} = \min(C_i^{max}, p_{i,s}^{train} - p_{i,s}^{off} + p_{i,s}^{station}) \quad (14)$$

3.5. Objective Function

The waiting time for all passengers(WAT) can be used to evaluate the operation strategy. It includes the waiting time of passengers who are taken to their destination and the passengers who are left in the station, which means that passengers who cannot be taken by the last train become a penalty factor of this problem. Thus, the operation strategy

should balance taking more passengers and faster arrival. Based on Equations (1)–(14), the objective function can be formulated below:

$$\begin{aligned}
 WAT = \sum_{i=0}^I \sum_{s=0}^{S-1} & \left[p_{i,s}^{left} (d_{i,s} - d_{i-1,s}) + \sum_{t=d_{i-1,s}+1}^{d_{i,s}} \lambda_s(t) (d_{i,s} - t + 1) \right] \\
 & + \sum_{s=0}^{S-1} \left[p_{I,s}^{left} (T - d_{I,s}) + \sum_{t=d_{I,s}+1}^T \lambda_s(t) (T - t + 1) \right] \tag{15}
 \end{aligned}$$

In real-time operation, passenger flow’s average entering speed will keep changing; the operation strategy needs to be optimised continually. However, it is impossible to adjust the station stop strategies that have already passed by the train. Thus, to optimise the operation based on detected variations in real-time, the departure time generated should be compared with the detecting time. If the departure time is earlier than or equal to the detecting time, it could not be optimised, and these departure times will be regarded as parameters in the objective function in future optimisation. Otherwise, the departure time is still a variable that could be optimised in real-time:

$$d_{i,s}^{real-time} = \begin{cases} d_{i,s}^{old} & t_D \leq d_{i,s}^{old} \\ d_{i,s}^{new} & t_D > d_{i,s}^{old} \end{cases} \tag{16}$$

As the constraints need to be met to guarantee the safety of the system, the optimisation function of RPOM based on nonlinear integer programming can be given as follows:

$$\left\{ \begin{aligned}
 \min WAT = \sum_{i=0}^I \sum_{s=0}^{S-1} & [p_{i,s}^{left} (d_{i,s}^{real-time} - d_{i-1,s}^{real-time}) + \\
 & \sum_{t=d_{i-1,s}^{real-time}+1}^{d_{i,s}^{real-time}} \lambda_s(t) (d_{i,s}^{real-time} - t + 1)] + \\
 & \sum_{s=0}^{S-1} [p_{I,s}^{left} (T - d_{I,s}^{real-time}) + \\
 & \sum_{t=d_{I,s}^{real-time}+1}^T \lambda_s(t) (T - t + 1)] \\
 s.t. & \quad \forall d_{-1,s} = 0; \forall l_{-1,s} = 0 \\
 & \quad \text{constraints (1) – (14), (16)}
 \end{aligned} \right. \tag{17}$$

4. Algorithm

From last section, the formulation of real-time passenger flow-oriented metro operation without timetables has been proposed. There are three requirements for the solving method. First, the operation strategies need to be adjusted and constantly based on optimising the result in real-time. Thus, the solving method needs to be fast enough. Second, the average passenger entering speed is a dynamic parameter. Third, the number of decision variables changes with the detecting time period.

Thus, to meet the requirements, a modified genetic algorithm, GA_RPOM, integrated with a macroscopic metro simulator, is proposed to solve the RPOM problem.

4.1. Macroscopic Metro Simulator

The four sections of the simulator will be introduced separately below. Based on this simulator, the locations of trains and passenger flow distribution could be tracked at any specific time in the research time horizon.

4.1.1. Metro Infrastructure Section and Passenger Flow Section

This section mainly focuses on data input and the descriptions of the passenger flow and the metro system. For passenger flow, the most critical data is the OD matrix, which

describes passengers' journeys between specific stations. For metro infrastructure, the input data mainly focus on the constraint conditions in the last section, which contains the running time, departure interval levels, different dwell time levels, and train capacity.

4.1.2. Timetable Section

Based on different departure intervals and dwell levels settled in the metro system section, each train has many optional combinations of interval and dwell time choices. This section assembles these combinations and generates, modifies, and records all trains' arrival times and departure times based on these combinations. The process of the timetable section is shown in Algorithm 1 below:

Algorithm 1: Timetable generator

Input: Departure interval $ll_{i,i-1}$ and dwell level $ld_{i,s}$ choice for trains in the time horizon

Output: Arrival and departure time

```

1       for train  $i \in \bar{I}$  do
2           for station  $s \in \bar{S}$  do
3               if  $i = 0$  and  $s = 0$ 
4                    $a_{i,s} = 0$ 
5                    $d_{i,s} = a_{i,s} + \omega ld_{i,s}$ 
6               else if  $i = 0$  and  $s \neq 0$ 
7                    $a_{i,s} = d_{i,s-1} + r_{i,(s,s+1)}^{ope}$ 
8                    $d_{i,s} = a_{i,s} + \omega ld_{i,s}$ 
9               else if  $i \neq 0$  and  $s = 0$ 
10                   $a_{i,s} = d_{i-1,s} + h ll_{i,i-1}$ 
11                   $d_{i,s} = a_{i,s} + \omega ld_{i,s}$ 
12              else if  $d_{i-1,s} - d_{i,s-1} - r_{i,(s,s+1)}^{ope} \geq h_{min}$ 
13                   $a_{i,s} = d_{i,s-1} + r_{i,(s,s+1)}^{ope}$ 
14                   $d_{i,s} = a_{i,s} + \omega ld_{i,s}$ 
15              else
16                   $a_{i,s} = d_{i,s-1} + h_{min}$ 
17                   $d_{i,s} = a_{i,s} + \omega ld_{i,s}$ 
18                  return  $a_{i,s}, d_{i,s}$ 
19              end if
20          end for
21      end for

```

4.1.3. Passenger–Train Interaction Section

This part mainly concentrates on the interaction between the metro system and the passenger flow; it calculates all passengers' waiting time and evaluates the system's capability. Trains are dispatched based on the strategies generated in the last section, and passengers arrive at each station according to the average passenger flow entering speed recorded by the passenger flow section. With trains departing and stopping based on the chosen departure interval and dwell time, passengers get off firstly to make more space for passengers preparing to board. The metro infrastructure section should define train capacity. If there is free space on the train, passengers will board in sequence according to different destinations until the train's maximum capacity is filled. Remaining passengers will keep waiting in the station. The process of the passenger–train interaction simulator is shown in Algorithm 2 below:

Algorithm 2: Passenger–train interaction simulator

Input: Passenger data from passenger flow section; system data from metro infrastructure section; departure and arrival time from timetable section

Output: Total passengers' waiting time WAT.

```

1      WAT = 0
2      for time t <= T do, t++
3          for station s ∈ S̄ do
4              WAT = WAT + pi,sstation
5          end for
6          for train i ∈ Ī do
7              t do
8                  if ai,s = t
9                      for pi,soff > 0 do
10                         pi,strain-
11                         ci,s++
12                     end for
13                 end if
14                 if di,s = t
15                     pi,sleft = pi,sstation
16                     for pi,sstation > 0 && ci,s > 0 do
17                         pi,son++
18                         pi,sleft-
19                     ci,s-
20                 end for
21             else
22                 continue
23             end if
24         end for
25     end for
26 end for
27 return WAT

```

4.2. GA_RPOM Algorithm

GA (genetic algorithm) is a metaheuristic inspired by the process of natural selection that belongs to the larger class of evolutionary algorithms (EA). GA is commonly used to generate high-quality solutions to search and optimise problems by relying on bio-inspired operators. In the GA, a population of candidate solutions (individuals, creatures, or phenotypes) to an optimisation problem evolves toward better solutions. Each candidate solution has a set of properties (its chromosomes or genotype) which can be mutated and altered; traditionally, solutions are represented in binary as strings of 0 s and 1 s.

Because of its clear concept, GA has become increasingly popular in the optimal train operational problems. For the RPOM problem, with the binary coding, the GA is more suitable to simulate the choices of dwell time and departure time interval in the metro system than other heuristic algorithms (such as the ant colony algorithm). With a large number of variables in the objective function, the GA could calculate the optimised result in a shorter time and also jump from the local optimal solutions with the crossover and the mutation process easily. Furthermore, the GA can be easily modified and integrated with the simulator for this problem. Based on the traditional GA, the modified process of GA_RPOM can be shown with the following steps:

Initialize population: in general, the population of a GA consists of fixed-length binary arrays called gene sequences which represent the specific variables in the objective functions. At first, the population will be generated randomly according to the designed population size.

For this research, we use binary numbers to represent the chosen departure time interval's level and dwelling time's level. Thus, each train can obtain a unique binary array

representing its operation strategy. After integrating different trains' binary arrays, an enormous binary array is generated as the gene sequence.

However, different from the traditional GA, the number of adjustable departure time intervals and dwelling times keeps changing. By integrating the macroscopic simulator, whenever the strategy needs to be optimised, the best gene sequence generated in the last detecting period would be recorded and compared with the new detecting time in the metro infrastructure section to modify the gene sequence length in real-time. The process to generate the gene sequence can be shown in Figure 3.

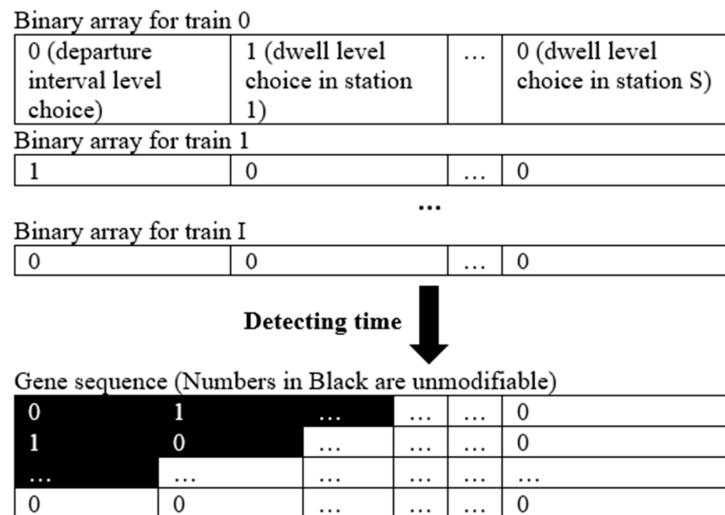


Figure 3. Approach to generate gene sequence for GA_RPOM.

Evaluation: This step evaluates all the gene sequences from the population generated in the previous stage. Typically, the evaluation process is based on the objective function. First, the binary arrays of gene sequence would be transformed into the variables in the objective function. Then, the evaluation indicators mark a better result with higher (or lower) scores.

In this research, the timetable section of the metro simulator could transform the binary arrays into specific operation strategies and ensure the objective function's constraints. The passenger–train interaction section of the simulator could record the waiting time for all passengers, which could be regarded as the evaluation indicator.

Selection: The gene sequence with a higher score is adopted from this step. Generally, a roulette algorithm is applied to prove the fairness of selection. Then, selected gene sequences will become parent gene sequences for the next step.

Crossover: In this step, all the parent gene sequences selected in the last step will mate (cross a part of binary arrays) to produce new baby gene sequences which inherit a part of their features, as shown in Figure 4.

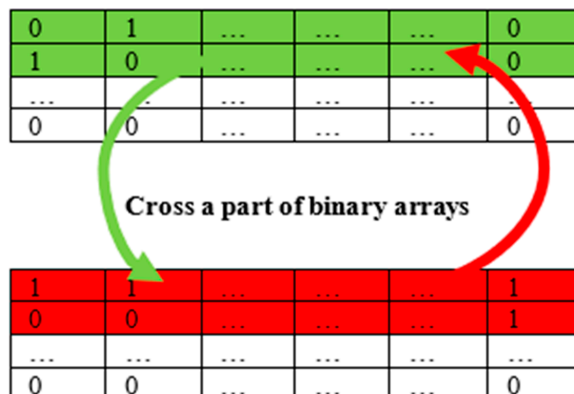


Figure 4. Crossover process.

Mutation: A part of the baby gene sequences generated in the last step is mutated. Based on the mutation rate, a part of the gene sequence will switch to opposite numbers (0 to 1, 1 to 0). The combination of mutating gene sequences constitutes the new population for the next period. The flow chart of GA_RPOM in the optimisation process and the interaction with the simulator is shown in Figure 5.

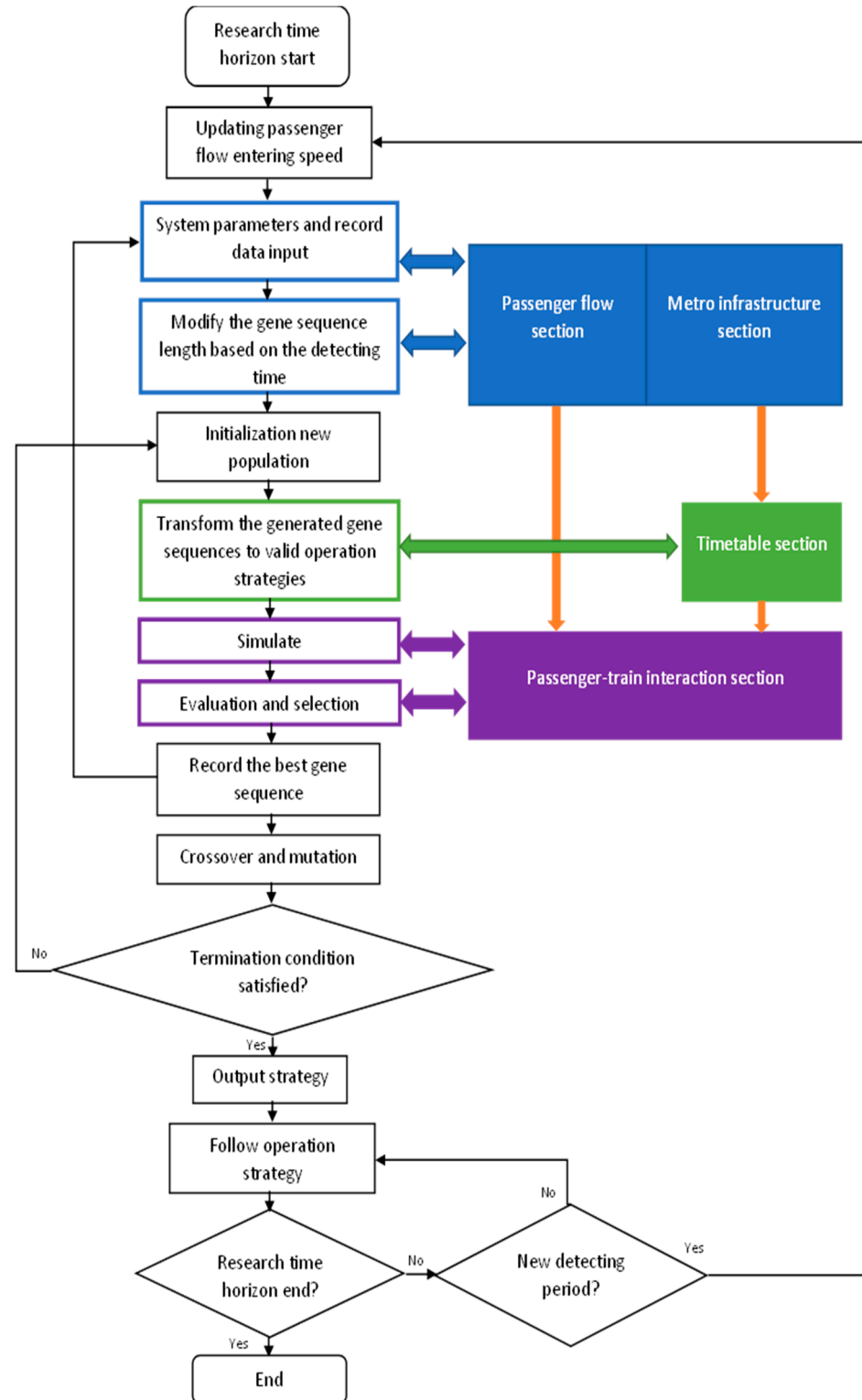


Figure 5. Flow chart of GA_RPOM in the optimisation process.

With massive experiments, GA_RPOM could solve most metro operation problems in 30 s. Based on this algorithm, metro systems' operation strategy could be kept generating and optimising in real-time.

5. Case Study

For this research, a case study based on Beijing Metro Line 19 is used to demonstrate the new operation method and evaluate the performance of GA_RPOM. The total length of the Beijing Metro Line 19 is 22.4 km with ten stations. The speed limit for the whole line is 120 km/h, and this line will be allocated with new 5G network communication and face recognition technology from Beijing Infrastructure Investment Co Ltd, Beijing, China. Before passengers enter the platforms, they need to choose their origin and destination from the ticket machines. Then, when they pass the platforms' gates, an Automated Fare Collection (AFC) system can collect and generate real-time OD data. Furthermore, the EUHT communication units will be set up in all stations to transfer the OD data. Generally, this detecting and transferring process will take 15 min. Thus, the 15 min is regarded as the detecting period. The case studies use a single-direction line from Mudanyuan to Xingong, whose infrastructure data is shown in Table 4. In macroscopic terms, the simplified line information and running time between stations are shown in Figure 6.

Table 4. Infrastructure parameters of Beijing Metro Line 19.

Parameter	Data
Number of trains	9
Number of stations	10 stations
Minimum interval	2 min
Research time horizon	90 min
Capacity of each train	2000
Optional dwell time	Short dwelling: 0.5 min, Long dwelling: 1.5 min
Optional departure interval	Short interval: 4 min, Long interval: 5 min
The number of solutions	2^{89}

Because this metro line is still under construction, the passenger data in the case studies is mocked and based on the early-stage desk research of the metro operator and the pre-designed capacity target of this line with the consideration of its connections with other commuting systems. Some extreme cases are also applied for testing. The passenger entering speed OD data is shown in Figure 7. The vertical axis represents different OD pairs, while the horizontal axis represents passenger entering speed; different colours represent the different detecting periods.

The case study compares the quality of service based on all passengers' waiting time in the research horizon between two traditional periodic timetables and the real-time passenger-oriented operation method presented in this paper. From Table 3, there are two optional dwelling times and two optional departure intervals; one periodic timetable is applied with the short dwelling time and short departure interval, named 'short periodic timetable', and the other one uses the long dwelling time and long departure interval, named 'long periodic timetable'. Contrarily, the RPOM method will keep generating operation strategies and optimising based on detecting passenger flow data in real-time.

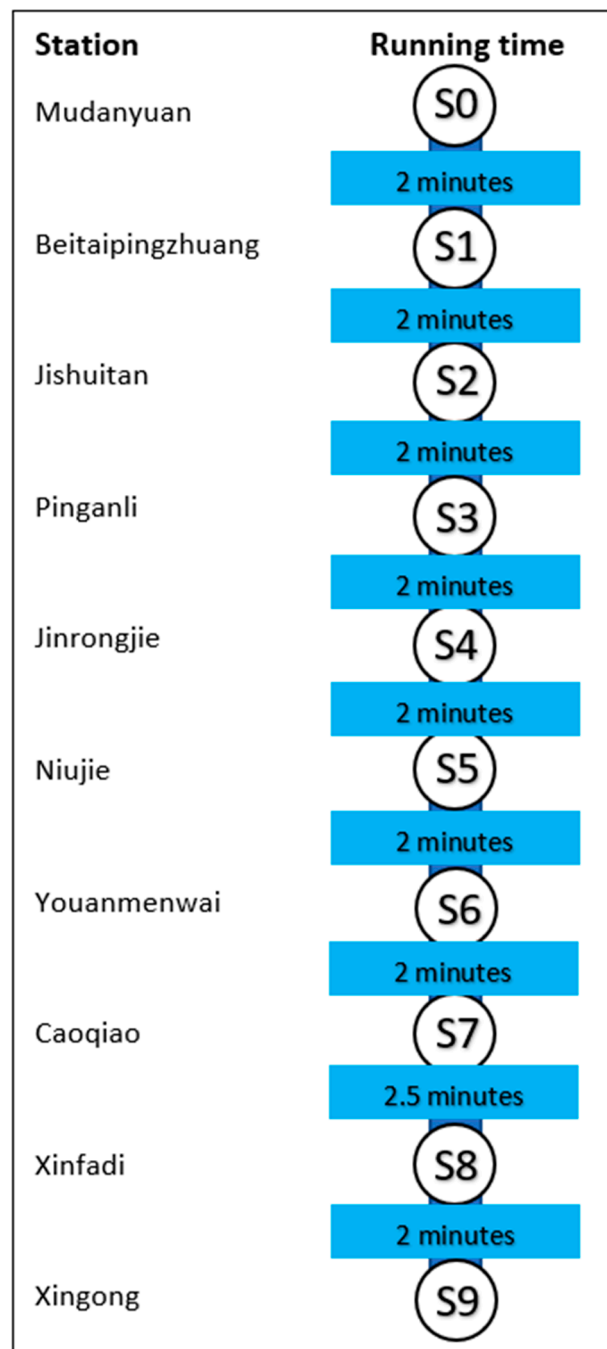


Figure 6. Simplified line information and the running time of Beijing Metro Line 19.

The case study was carried out on a Windows 10 laptop with a Core i5 8th generation CPU and 8 GB of RAM. The GA_RPOM program is coded in the Java language based on JDK 1.8. The population size is 200, the mutation rate is 0.2 with a maximum mutation number of 5, and the maximum iteration number is 600. The parameters of the GA_RPOM for this case study are shown in Table 5. We can obtain the operation strategies in 30 s based on the algorithm program. The convergence graph of the GA_RPOM based on the dynamic passenger flow data in the research time horizon is shown in Figure 8. The operation strategies based on RPOM and the comparison of all passenger waiting times between RPOM and periodic timetables are shown in Table 6.

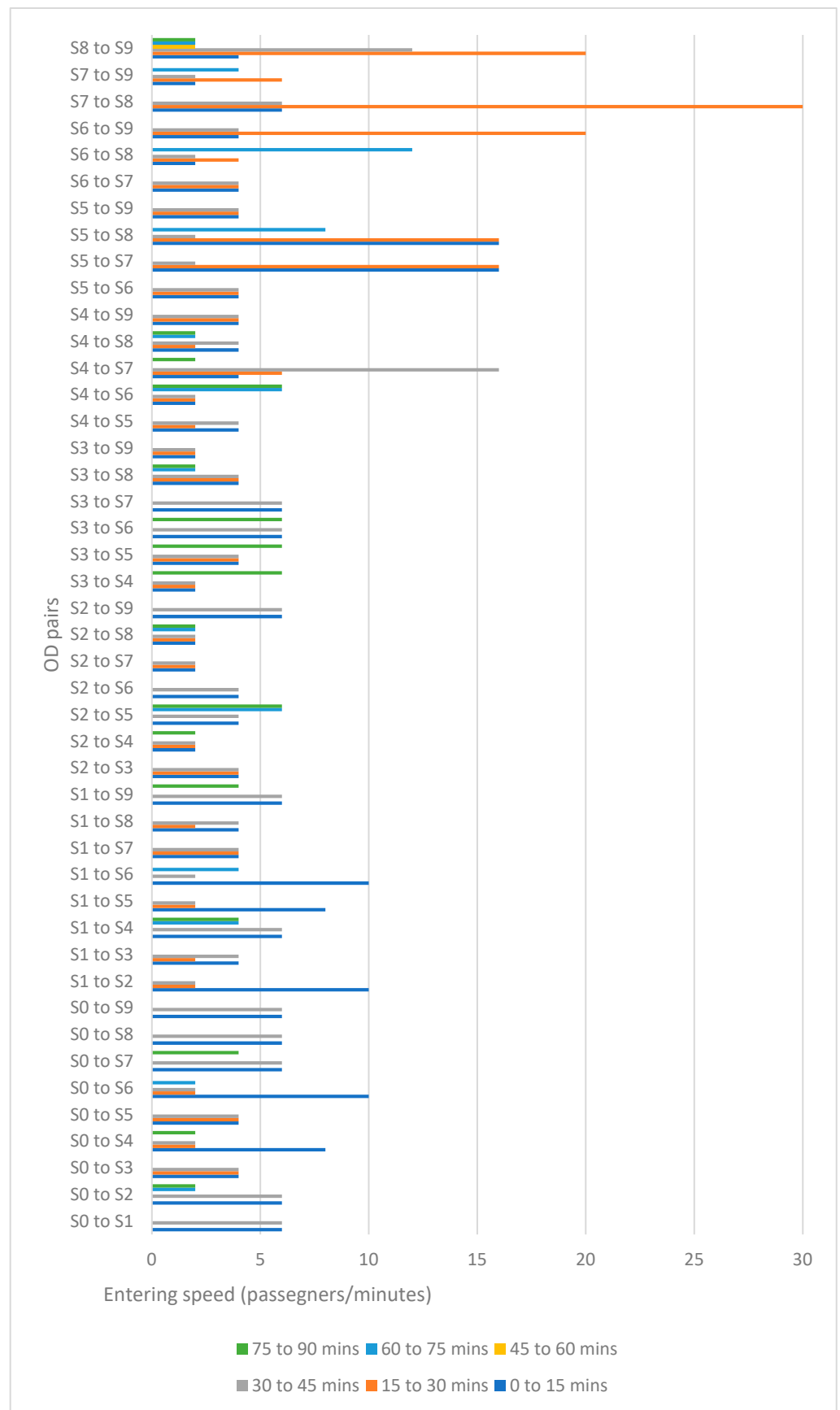


Figure 7. Passenger entering speed OD data.

Table 6. Cont.

Detecting time		60 to 75 min								
Strategy		RPOM			ST			LT		
All passengers waiting time (minutes)		107,819			120,759			117,163		
	DI	DW1	DW2	DW3	DW4	DW5	DW6	DW7	DW8	DW9
T1	X	L	S	S	S	S	S	S	S	L
T2	L	L	S	L	S	S	S	S	L	L
T3	L	L	L	S	L	L	S	L	S	L
T4	L	L	L	S	L	L	L	S	L	S
T5	L	L	S	L	L	L	L	S	S	L
T6	L	L	S	L	L	S	S	L	L	L
T7	L	L	S	S	S	L	L	S	L	L
T8	S	S	S	L	L	L	L	L	L	L
T9	L	L	L	L	L	L	L	L	L	L
Detecting time		75 to 90 min								
Strategy		RPOM			ST			LT		
All passengers waiting time (minutes)		108,416			121,194			117,920		
	DI	DW1	DW2	DW3	DW4	DW5	DW6	DW7	DW8	DW9
T1	X	L	S	S	S	S	S	S	S	L
T2	L	L	S	L	S	S	S	S	L	L
T3	L	L	L	S	L	L	S	L	S	L
T4	L	L	L	S	L	L	L	S	L	S
T5	L	L	S	L	L	L	L	L	S	S
T6	L	L	S	L	L	S	S	S	L	L
T7	L	L	S	S	S	L	L	S	L	L
T8	S	S	S	L	L	L	L	L	L	L
T9	L	L	L	L	L	L	L	L	L	L

Abbreviation: DW + number: dwelling time + station number; DI: departure interval; T + number: train's number; S: short time strategy; L: long time strategy; ST: short periodic timetable; LT: long periodic timetable. Annotation: The strategies in the White box are modifiable; The strategies in the Grey box are fixed.

From the above table, RPOM could optimise and modify any later operation strategy for both dispatched trains and follow-up trains at the beginning of every detecting period. In the 90 min research time horizon, the operation strategies' optimisation process has been implemented six times, or 53 dwelling times, and the departure intervals have been rescheduled. Because the waiting time of passengers who have been left is also counted, from time 0 to 45 min, RPOM set each train's departure interval to 'long' to take more passengers, and it kept adjusting the strategies to lead trains to stop longer in the station with high passenger flow entering speed. Between 45 and 60 min, an extreme situation happened: only two passengers entered every minute in the last station. Thus, RPOM changes the strategy of Train 8 rapidly to a short departure interval. After 60 min, it kept the long dwelling strategies to take more passengers to their destination. Typically, these modifications were implemented after noticeable passenger flow variations occurred, especially when a sudden, massive passenger flow occurred. Unlike the fixed dwelling times strategies, with the GA_RPOM method, trains aim to dwell shorter in the stations with lower passenger entering speeds and stop longer in the stations with high passenger entering speeds. Furthermore, GA_RPOM keeps modifying its operation strategies, making the massive passenger flows in the busy stations embark earlier and preventing these passengers from being stranded in these stations in real-time. With high flexibility, in most situations, the performance of RPOM was better than the periodic timetables, which is reflected by all of the passenger waiting time data. Though in some extreme situations, with the number of modifiable strategies decreased, the performance of RPOM was slightly worse than the short periodic timetable, which occurred once from 45 to 60 min, the

overall performance of RPOM is much better than two periodic timetables. Moreover, as passenger flow variations are always in real-life operation, a flexible operation method is more valuable than a perfectly prepared fixed timetable.

With a fixed number of trains, compared with the periodic strategies such as the ST and LT, there is no significant increase in trains' running costs of GA_RPOM, and it is easily applied in practice. Of course, extra operational training for the metro operators and drivers is necessary. However, compared with the experience-based operation method, more accurate indicators could be provided to the operators and drivers by GA_RPOM.

6. Conclusions

In this paper, an innovative passenger flow-oriented metro operation method without timetables has been presented. A mathematical model is proposed to minimise all passenger waiting times with a limited rolling stock number in the research horizon. A new algorithm program based on a GA, GA_RPOM, integrated with a simulator, has been developed to solve the model and optimise the metro operation plan in an acceptable computational time. Different from the traditional GA for other dynamic scheduling problems, GA_RPOM could modify the gene sequence length to optimise the operation strategies for the metro system in real-time. A case study based on the Beijing Metro Line 19 has been carried out to validate the effectiveness of the proposed method. Compared with traditional periodic timetables, the passenger waiting time has been reduced in most scenarios and the flexibility of the operation has been significantly improved.

The limitations found for this research were that no real-life passenger flow data were collected for the test, as the Beijing Metro Line is still under construction, and no other test metro systems were found for the proposed optimisation method.

In future works, we aim to design more test instances for the proposed method and also to improve the computing time by integrating the genetic algorithm with a machine learning method such as the decision tree algorithm.

Author Contributions: Conceptualisation, L.H. and L.C.; methodology, L.H. and L.C.; software, L.H.; validation, L.H., L.C., S.Y. and X.F.; formal analysis, L.H. and L.C.; investigation, L.H. and L.C.; writing—original draft preparation, L.H.; writing—review and editing, L.C. and J.L.; supervision, L.C. and C.R.; project administration, L.C. and C.R.; funding acquisition, L.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by China Urban Sustainable Transport Research Centre under Grant No.2018YFC0809900 and supported by project "Research and Development of Intelligent Metro Network Operation Decision Support System" by Hefei Metro.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cacchiani, V.; Toth, P. Nominal and robust train timetabling problems. *Eur. J. Oper. Res.* **2012**, *219*, 727–737. [[CrossRef](#)]
2. Cacchiani, V.; Furini, F.; Kidd, M.P. Approaches to a real-world Train Timetabling Problem in a railway node. *Omega* **2016**, *58*, 97–110. [[CrossRef](#)]
3. Niu, H.; Zhou, X. Optimising urban rail timetable under time-dependent demand and oversaturated conditions. *Transp. Res. Part C* **2013**, *36*, 212–230. [[CrossRef](#)]
4. Barrena, E.; Canca, D.; Coelho, L.; Laporte, G. Single-line rail rapid transit timetabling under dynamic passenger demand. *Transp. Res. Part B Methodol.* **2014**, *70*, 134–150. [[CrossRef](#)]
5. Higgins, A.; Kozan, E.; Ferreira, L. Optimal scheduling of trains on a single line track. *Transp. Res. Part B* **1996**, *30*, 147–161. [[CrossRef](#)]
6. Cacchiani, V.; Huisman, D.; Kidd, M.; Kroon, L.; Toth, P.; Veelenturf, L.; Wagenaar, J. An overview of recovery models and algorithms for real-time railway rescheduling. *Transp. Res. Part B Methodol.* **2014**, *63*, 15–37. [[CrossRef](#)]
7. Corman, F.; Quaglietta, E. Closing the loop in real-time railway control: Framework design and impacts on operations. *Transp. Res. Part C Emerg. Technol.* **2015**, *54*, 15–39. [[CrossRef](#)]

8. Corman, F.; D'Ariano, A.; Pacciarelli, D.; Pranzo, M. Bi-objective conflict detection and resolution in railway traffic management. *Transp. Res. Part C* **2012**, *20*, 79–94. [[CrossRef](#)]
9. Yang, X.; Chen, A.; Li, X.; Ning, B.; Tang, T. An energy-efficient scheduling approach to improve the utilization of regenerative energy for metro systems. *Transp. Res. Part C* **2015**, *57*, 13–29. [[CrossRef](#)]
10. Jin, B.; Sun, P.; Wang, Q.; Feng, X. Two-step method to reduce metro transit energy consumption by optimising speed profile and timetable. *IET Intell. Transp. Syst.* **2020**, *14*, 1097–1107. [[CrossRef](#)]
11. Zhou, X.; Zhong, M. Bicriteria train scheduling for high-speed passenger railroad planning applications. *Eur. J. Oper. Res.* **2005**, *167*, 752–771. [[CrossRef](#)]
12. Niu, H.; Zhou, X.; Gao, R. Train scheduling for minimizing passenger waiting time with time-dependent demand and skip-stop patterns: Nonlinear integer programming models with linear constraints. *Transp. Res. Part B Methodol.* **2015**, *76*, 117–135. [[CrossRef](#)]
13. Carey, M.; Crawford, I. Scheduling trains on a network of busy complex stations. *Transp. Res. Part B Methodol.* **2007**, *41*, 159–178. [[CrossRef](#)]
14. Cordone, R.; Redaelli, F. Optimising the demand captured by a railway system with a regular timetable. *Transp. Res. Part B* **2011**, *45*, 430–446. [[CrossRef](#)]
15. Li, X.; Lo, H.K. An energy-efficient scheduling and speed control approach for metro rail operations. *Transp. Res. Part B Methodol.* **2014**, *64*, 73–89. [[CrossRef](#)]
16. Sun, L.; Jin, J.G.; Lee, D.-H.; Axhausen, K.W.; Erath, A. Demand-driven timetable design for metro services. *Transp. Res. Part C Emerg. Technol.* **2014**, *46*, 284–299. [[CrossRef](#)]
17. Yin, J.; Chen, D.; Yang, L.; Tang, T.; Ran, B. Efficient real-time train operation algorithms under uncertain passenger demands. *IEEE Trans. Intell. Transp. Syst.* **2016**, *17*, 2600–2612. [[CrossRef](#)]
18. Yin, J.; Yang, L.; Tang, T.; Gao, Z.; Ran, B. Dynamic passenger demand oriented metro train scheduling with energy-efficiency and waiting time minimization: Mixed-integer linear programming approaches. *Transp. Res. Part B Methodol.* **2017**, *97*, 182–213. [[CrossRef](#)]
19. Halim, H.I.; Sakr, M.; Aly, W.M. Metro timetable optimization from passenger perspective based on simulation models and incomplete data of passenger flow. In Proceedings of the 2016 IEEE Symposium Series on Computational Intelligence (SSCI), Athens, Greece, 6–9 December 2016; pp. 1–9. [[CrossRef](#)]
20. Petersen, E.R.; Taylor, A.J.; Martland, C.D. An introduction to computer-assisted train dispatch. *J. Adv. Transp.* **1986**, *20*, 63–72. [[CrossRef](#)]
21. Ghoseiri, K.; Szidarovszky, F.; Asgharpour, M.J. A multi-objective train scheduling model and solution. *Transp. Res. Part B Methodol.* **2004**, *38*, 927–952. [[CrossRef](#)]
22. Li, X.; Wang, D.; Li, K.; Gao, Z. A green train scheduling model and fuzzy multi-objective optimisation algorithm. *Appl. Math. Model.* **2013**, *37*, 617–629. [[CrossRef](#)]
23. D'Ariano, A.; Pranzo, M.; Hansen, I.A. Conflict resolution and train speed coordination for solving real-time timetable perturbations. *IEEE Trans. Intell. Transp. Syst.* **2007**, *8*, 208–222. [[CrossRef](#)]
24. Chen, L. Real Time Traffic Management in Junction Areas and Bottleneck Sections on Mainline Railways. Ph.D. Thesis, University of Birmingham, Birmingham, UK, 2012.
25. Dorfman, M.; Medanic, J. Scheduling trains on a railway network using a discrete event model of railway traffic. *Transp. Res. Part B Methodol.* **2004**, *38*, 81–98. [[CrossRef](#)]
26. Mu, S.; Dessouky, M. Efficient dispatching rules on double tracks with heterogeneous train traffic. *Transp. Res. Part B Methodol.* **2013**, *51*, 45–64. [[CrossRef](#)]
27. Liu, J.; Chen, L.; Roberts, C.; Nicholson, G.; Ai, B. Algorithm and peer-to-peer negotiation strategies for train dispatching problems in railway bottleneck sections. *IET Intell. Transp. Syst.* **2019**, *13*, 1717–1725. [[CrossRef](#)]
28. Yang, L.; Zhou, X.; Gao, Z. Credibility-based rescheduling model in a double-track railway network: A fuzzy reliable optimisation approach. *Omega* **2014**, *48*, 75–93. [[CrossRef](#)]
29. Zhan, S.; Kroon, L.G.; Veelenturf, L.P.; Wagenaar, J.C. Real-time high-speed train rescheduling in case of a complete blockage. *Transp. Res. Part B Methodol.* **2015**, *78*, 182–201. [[CrossRef](#)]
30. Gao, Y.; Kroon, L.; Schmidt, M.; Yang, L. Rescheduling a metro line in an over-crowded situation after disruptions. *Transp. Res. Part B Methodol.* **2016**, *93*, 425–449. [[CrossRef](#)]
31. Fu, L.; Liu, Q.; Calamai, P. Real-time optimisation model for dynamic scheduling of transit operations. *Transp. Res. Rec.* **2003**, *1857*, 48–55. [[CrossRef](#)]
32. Eberlein, X.J.; Wilson, N.H.M.; Bernstein, D. The Holding Problem with Real-Time Information Available. *Transp. Sci.* **2001**, *35*, 1–18. [[CrossRef](#)]
33. Ghoneim, N.; Wirasinghe, S. Optimum zone structure during peak periods for existing urban rail lines. *Transp. Res. Part B Methodol.* **1986**, *20*, 7–18. [[CrossRef](#)]
34. Ceder, A. Optimal design of transit short-turn trips. *Transp. Res. Rec.* **1989**, *1221*, 8–22.
35. Delle Site, P.; Filippi, F. Service optimisation for bus corridors with short-turn strategies and variable vehicle size. *Transp. Res. Part A* **1998**, *32*, 19–38.

36. Eberlein, X.J.; Wilson, N.H.; Barnhart, C.; Bernstein, D. The real-time deadheading problem in transit operations control. *Transp. Res. Part B Methodol.* **1998**, *32*, 77–100. [[CrossRef](#)]
37. Chen, L.; Roberts, C.; Schmid, F.; Stewart, E. Modeling and solving real-time train rescheduling problems in railway bottleneck sections. *IEEE Trans. Intell. Transp. Syst.* **2015**, *16*, 1896–1904. [[CrossRef](#)]
38. Gao, Y.; Yang, L.; Gao, Z. Real-time automatic rescheduling strategy for an urban rail line by integrating the information of fault handling. *Transp. Res. Part C Emerg. Technol.* **2017**, *81*, 246–267. [[CrossRef](#)]
39. Hou, Z.; Gao, S.; Nicholson, G.; Chen, L.; Chen, F.; Roberts, C.; Dong, H. A Hybrid Intelligent Algorithm for Real Time Metro Traffic Regulation in Cases of Disturbance. In Proceedings of the 2018 21st International Conference on Intelligent Transportation Systems (ITSC), Maui, HI, USA, 4–7 November 2018; pp. 3929–3934.