SCIENCE CHINA
Information Sciences

• **Supplementary File** •

A Cuckoo Search Approach for Automatic Train Regulation under Capacity Limitation

Zhuopu HOU[1], Min ZHOU[1], Clive ROBERTS[2] & Hairong DONG[1*]

[1] *State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing* 100044*, China;*
[2] *School of Electronic, Electrical and Systems Engineering, University of Birmingham, Birmingham* B15 2TT*, United Kingdom*

Appendix A

Appendix A.1 Parameters and notation

In order to conveniently demonstrate the automatic train regulation problem in cases of disturbances and describe the proposed optimization model, the relevant indices parameters and variables are listed in Table A1. We consider a metro single line in this research as shown in Fig. A1, where stations are numbered as 1, 2, $\cdots$, $|I|$ and inter-stations are numbered as 1, 2, $\cdots$, $|I|-1$.

Table A1 Parameters and Variables

i	Index of the metro station, where $i \in \{1, 2, ..., \lvert I\rvert\}$
j	Index of the metro train, where $j \in \{1, 2, ..., \lvert J\rvert\}$
$l_{i,j}$	The number of left-behind passengers on the platform when train j departs from station i
$w_{i,j}$	The number of waiting passengers on the platform before train j arriving at station i
$f^{\mathrm{t}}_{i,j}$	The free capacity of train j when it dwells at station i
$o_{i,j}$	The number of in-vehicle passengers before train j arriving at station i
$a_{i,j}$	The number of alighting passengers when train j dwells at station i
$b_{i,j}$	The number of boarding passengers when train j dwells at station i
$e_{i,j}$	The number of passengers who are allowed to enter the platform before train j arriving at station i
$f^{\mathrm{p}}_{i,j}$	The free capacity of the platform before train j arriving at station i
$s_{i,j}$	The number of passengers who are blocked outside the platform before train j arriving at station i
$m_{i,j}$	The number of passengers who want to enter the platform before train j arriving at station i
$n_{i,j}$	The number of passengers who arrive at station i during a departure interval
$R^{\mathrm{u}}_{i-1,j}$	The upper bound of the running time of train j in section $i-1$, $i \in \{2, ..., \lvert I\rvert\}$
$R^{\mathrm{l}}_{i-1,j}$	The lower bound of the running time of train j in section $i-1$, $i \in \{2, ..., \lvert I\rvert\}$
$\lambda_{i,j}$	The passenger arrival rate at station i during one departure interval
$\omega_{i,j}$	The alighting ratio of train j at station i
CT	The maximum allowable capacity of the metro train
CP	The maximum allowable capacity of the platform
$W^{\mathrm{l}}_{i,j}$	The lower bound of the dwell time of train j at station i
$W^{\mathrm{u}}_{i,j}$	The upper bound of the dwell time of train j at station i
H_1	The minimum safety interval between two trains in the same section
H_2	The minimum safety interval between two trains at the same station
$A_{i,j}$	The time when train j arrives at station i in planned schedule
$D_{i,j}$	The time when train j departs from station i in planned schedule
$x_{i,j}$	Decision variables, the actual time when train j arrives at station i
$y_{i,j}$	Decision variables, the actual time when train j departs from station i

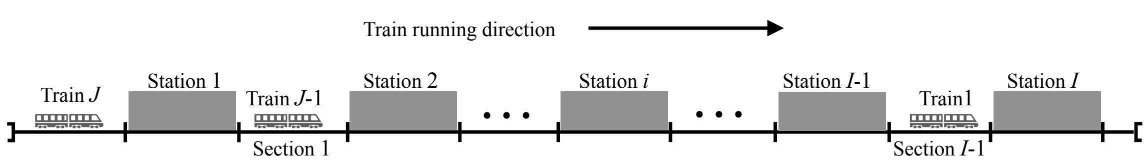

Figure A1 Typical metro line sketch

* Corresponding author (email: hrdong@bjtu.edu.cn)

Appendix A.2 Problem Assumptions

To support the rationality of the proposed model, we make the following assumptions.

1). The duration of the initial delay can be predicted accurately when the disturbance occurs. The impact of the disturbance on other connected lines is disregarded since we only focus a single line in this research.

2). The number of passengers who arrive at the station during one departure interval is assumed to be distributed uniformly [1]. We also do not consider the transfer passengers since a single metro line is considered in this paper.

3). The skip-stop pattern and train holding strategy is not considered in this paper. The meeting and overtaking operation of trains is not allowed in this study since there is no side track at the metro platform.

Appendix A.3 The passenger flow simulation model

When the metro train dwells at the station, the dynamic exchange of passenger flow occurs between the vehicle and the platform, that is, passengers waiting at the platform board the train, and passengers who need to get off the train enter the platform. Figure A2 illustrates the state of the passenger flow and trains at time T. The capacity of trains and platforms is limited, the passengers cannot enter the platform once the crowdedness degree of the platform reaches its allowable capacity, and not all the passengers on the platform can board the train. However, most previous studies [2–11]on train regulation/timetabling problems do not consider the capacity constraints of trains and platforms, which is inconsistent with the actual situation. In this section, we intend to build a detailed passenger flow simulation model to evaluate the objective of the proposed optimization problem.

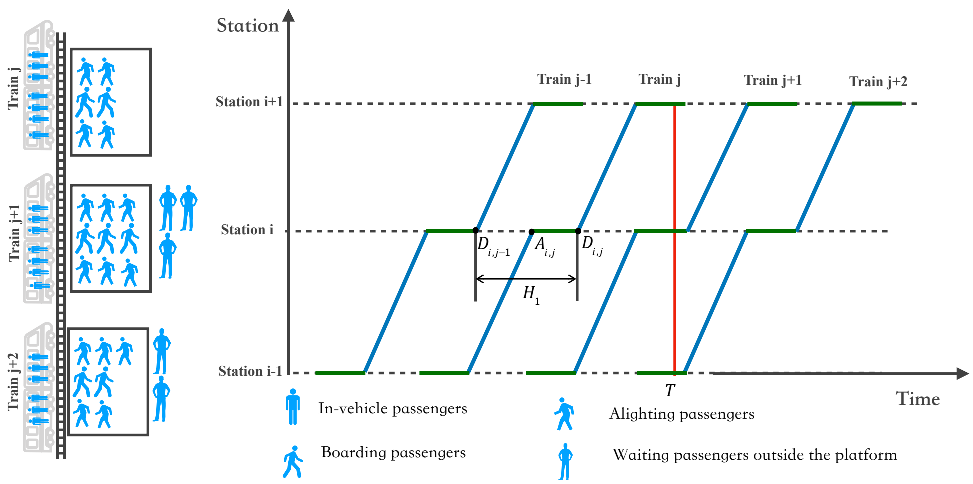

Figure A2 Illustration of the state of passenger flow and trains at time T

When passengers accumulate due to delays, the principal task of the dispatcher is to evacuate the passenger flow on the platform as soon as possible, that is, the fewer stranded passengers the better. The number of left-behind passengers when train j departs from station i depends on the number of waiting passengers and the free capacity of the train, which can be denoted as

$$l_{i,j} = \max\{(w_{i,j} - f^t_{i,j}), 0\}. \tag{A1}$$

The number of passengers waiting at station i before train j arriving consists of the left-behind passengers after the last train $i-1$ leaving and the newly-entered passengers on the platform. Thus, the number of waiting passengers on the platform can be expressed as

$$w_{i,j} = l_{i,j-1} + e_{i,j}. \tag{A2}$$

The free capacity of the train after passengers getting off it can be given by

$$f^{\mathrm{t}}_{i,j} = CT - o_{i,j} + a_{i,j}. \tag{A3}$$

The number of onboard passengers when train j arrives at station i is equal to that when the train left the previous station, which can be expressed as

$$o_{i,j} = o_{i-1,j} + b_{i,j} - a_{i,j}. \tag{A4}$$

When the train dwells at the station, the number of passengers who will alight from train j at station i is assumed to be proportional to the number of passengers in the train j [9], which can be expressed as

$$a_{i,j} = \omega_{i,j} \cdot o_{i,j}. \tag{A5}$$

The number of passengers who can board train j at station i can be given by

$$b_{i,j} = \min\{f^t_{i,j}, w_{i,j}\}. \tag{A6}$$

The number of allowable-entering passengers on the platform depends on the free capacity of the platform. The passengers on the platform include passengers alighted from the train, passengers waiting to board the train, and passengers who have just arrived at the platform. The sum of the number of these passengers is not allowed to exceed the maximum allowable capacity of the platform, otherwise, it will cause potential safety hazards. The number of passengers who can enter the platform is given by

$$e_{i,j} = \min\{f^{p}_{i,j}, m_{i,j}\}. \tag{A7}$$

Before the train j arriving at station i, passengers left behind by the former train $j-1$ are still on the platform. There also should be enough room on the platform for passengers who will get off train j. Thus the free capacity of the platform when train j arrives at station i can be expressed as

$$f^{\mathrm{p}}_{i,j} = CP - l_{i,j-1} - a_{i,j}. \tag{A8}$$

Due to the limited capacity of the platform, some new arrival passengers might be blocked outside the platform. The number of these passengers can be denoted as

$$s_{i,j} = \max\{(m_{i,j} - f^{p}_{i,j}), 0\}. \tag{A9}$$

The total number of the passengers who want to enter the platform when train j arrives at station i can be expressed as

$$m_{i,j} = s_{i,j-1} + n_{i,j}. \tag{A10}$$

Based on assumption 2, the number of new arrival passengers during a departure interval can be denoted as

$$n_{i,j} = \lambda_{i,j} \cdot (y_{i,j} - y_{i,j-1}). \tag{A11}$$

Appendix B

Cuckoo are famous for their aggressive procreation strategy that they lay eggs in the nest of other birds and clear away eggs of the host bird from the nest. In cuckoo search algorithm, a group of nests is regarded as a population, and the eggs in each nest are regarded as a feasible solution to the optimization problem. The cuckoo randomly chooses a nest to lay eggs as a new solution to replace the worse solution. At the same time, a certain proportion of the host bird's nest will be abandoned and replaced by a new solution created randomly.

As we all know, randomization plays an important role in the search mechanism of meta-heuristic algorithms, the essence of which is the random walk [12]. The Lévy flight is adopted in Cuckoo Search as the position update method of solutions, which is kind of an efficient random walk. Given a solution $\vec{x}$ for the nest i, the next position of it can be expressed as

$$\vec{x}_i(t+1) = \vec{x}_i(t) + \alpha \oplus L(u,v), \tag{B1}$$

where t is the index of the current iteration; α is a positive number which means the step size coefficient and should be related to the interests of the optimization problem; $\oplus$ is the XOR operator which means entry-wise multiplications; $L(u,v)$ stands for a Lévy flight. Currently, the use of Mantegna algorithm is one of the most effective methods to achieve Lévy flight [13], which can be given by

$$L(u,v) = \frac{u}{|v|^{\frac{1}{\beta}}}, \tag{B2}$$

where u and v both follow a normal distribution, which is

$$u \sim N(0, \sigma_u^2), \tag{B3}$$

$$v \sim N(0, \sigma_v^2), \tag{B4}$$

where

$$\beta \in [1,2], \tag{B5}$$

$$\sigma_v = 1, \tag{B6}$$

$$\sigma_u = \left\{ \frac{\Gamma(1+\beta)\sin(\pi\beta/2)}{\Gamma[(1+\beta)/2]\beta 2^{(\beta-1)/2}} \right\}^{1/\beta}. \tag{B7}$$

Next, we introduce the procedure of solving automatic train regulation problem by Cuckoo Search algorithm. The detailed procedures are described as follows.

Step 1 Initialize parameters, which consist of the nest population size N_{size}, the discovery probability P_a, the total number of iteration N_{iter}, the passenger flow information, the original schedule and the disturbance information.

Step 2 Solution encoding. In CS algorithm, eggs in each nest stands for a solution. The decision variables of the proposed model are the arrival time $x_{i,j}$ and the departure time $y_{i,j}$, which are regarded as the nest and encoded in decimal form. The structure of the nest is shown in Fig. B1.

Step 3 Generate the population. In order to improve the diversity of the initial population, the nest position $\vec{x}_i$ is initialized randomly, which follows normal distribution. Evaluate the fitness of each nest according to the objective function and the passenger flow model; find the current best nest k.

Step 4 Randomly get a cuckoo i. In other words, generate a new solution by Lévy flight and calculate its fitness value $F(i)$.

Step 5 Randomly select a nest j and calculate fitness value $F(j)$. Note that nest k cannot be selected. Compare the fitness cost of nest$i$ and nest$j$, keep the better one as a solution.

$$\begin{bmatrix} x_{i,j} & \cdots & x_{i,|J|} \\ \vdots & \ddots & \vdots \\ x_{|I|,j} & \cdots & x_{|I|,|J|} \end{bmatrix} \begin{bmatrix} y_{i,j} & \cdots & y_{i,|J|} \\ \vdots & \ddots & \vdots \\ y_{|I|,j} & \cdots & y_{|I|,|J|} \end{bmatrix}$$

Figure B1 The nest encoding structure

Step 6 Discovery and abandon. Generate a random number r, if $r < P_a$, find the worst nest whose fitness value is the largest, then update its position by Lévy flight.

Step 7 Solutions sort. Rank all solutions according to the fitness value, and record the best nest and its cost.

Step 8 Termination condition. When the maximum number of iterations N_{iter} is reached, return the optimal solution and its cost. Otherwise, go to *Step* 4 and continue to iterate.

Appendix C

In this section, we conduct several experiments based on the operation data of Beijing Subway Line 9 to demonstrate the proposed model and algorithm. The Beijing Subway Line 9 consists of 13 stations and 12 sections, which is also a busy metro system, the passenger flow in several stations is oversaturated during peak hours. Once a disturbance occurs, it is likely to cause the passenger flow to accumulate on platforms due to the service gap. Thus an effective regulation solution integrating the information of the dynamic passenger flow is essential to improve the operation and service quality. The simulation environment of case studies and the Cuckoo Search algorithm is coded in Matlab R2014a and run on a personal computer with 1.6 GHz Intel Core i5 processor and 8 GB RAM.

Normally, there are 21 trains operating on Beijing Subway Line 9. We take the train running from the National Library to Guogongzhuang as the experimental object. The control time window considered in case studies is from 7:30 to 9:00 in morning rush hours and the start time is set as 0 for the sake of convenience. The data of planned running time in each section and dwell time at each station is listed in TableC1. The scheduled departure interval is set as 180 s. Then the original timetable can be obtained, the corresponding train time-distance diagram is shown in Fig. C1. The trains are named train 1 to train 21 in order from left to right. The data of lower and upper bounds for the rescheduled dwell time and running time are listed in Table C2. The passenger arrival rate and alighting ratio can be estimated based on the historical passenger flow data from the Automatic Fair Collection (AFC) system and the statistical technique [5].The passenger flow data adopted in this research is from article [9].

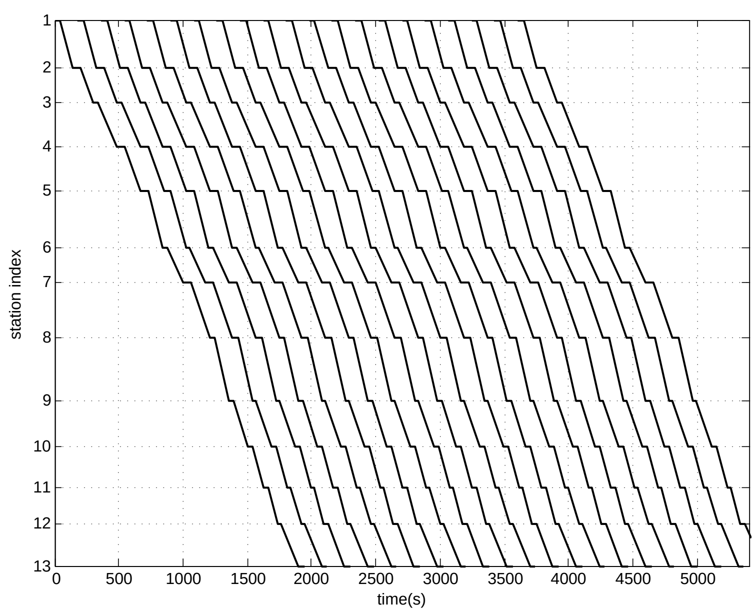

Figure C1 The original train diagram

Appendix C.1 Parameter tuning and validity verification for the algorithm

In this section, we use a typical delay scenario to validate the effectiveness of the proposed method. We set up several sets of parameters and select the best set as the parameters of the cuckoo algorithm to solve automatic train regulation problem. We assume that a medical emergency occurred when the eighth train dwells at the fifth station (Beijing West Railway Station), which causes a departure delay with the duration of 130 s. In other words, $s = 5$, $t = 8$ and $d_{s,t} = 130$.

Table C1 Planned schedule of Beijing Subway Line 9

Station Name	Station Index (i)	Dwell time (s)	Section Index (k)	Section length (m)	Running time (s)
National Library	1	45	1	1096	100
Baishiqiao South	2	60	2	1044	100
Baiduizi	3	35	3	1912	146
Military Museum	4	60	4	1398	122
Beijing West RS	5	60	5	1171	112
Liuliqiao East	6	35	6	1309	123
Liuliqiao	7	60	7	1779	147
Qilizhuang	8	45	8	1326	109
Fengtaidongdajie	9	30	9	1585	119
Fengtainanlu	10	35	10	980	87
keyilu	11	30	11	788	77
Fengtai SP	12	30	12	1348	128
Guogongzhuang	13	45	-	-	-

Table C2 Parameter of the operation data

Station	Minimum dwell time	maximum dwell time	section	minimum running time	maximum running time
1	30	65	1	94	120
2	40	75	2	92	120
3	20	55	3	136	166
4	40	75	4	114	142
5	40	75	5	104	132
6	20	50	6	112	143
7	40	75	7	135	167
8	35	60	8	105	129
9	20	50	9	108	139
10	20	55	10	81	107
11	20	50	11	72	97
12	20	50	12	120	148
13	30	55	-	-	-

The step size coefficient α is related to the scale of the optimization problem and the discovery rate P_a balance the local search and the randomization. Therefore, we need to select a set of suitable parameters (α, P_a) from several sets of candidate parameters to optimize the performance of the algorithm. The number of the nest N_{size} and the maximum number of iteration N_{iter} are set as 25 and 100, respectively. The weight of the total train delay and the number of stranded passengers are 0.5 and 0.5 respectively. The normalization values of two objectives are calculated based on a heuristic algorithm in paper [6]. The normalization value for the total train delay is 1901 s and that for the total stranded passengers is 1413 pax. Each group of tests runs independently 100 times and the effect of different parameter sets on algorithm performance is shown in Table C3, where the the success rate refers to the ratio of the number of times the cuckoo algorithm gets the optimal solution to the total number of simulations under a given parameter set. As shown in Table C3, we conducted 9 groups of tests, and parameters of the fourth test performed best. Thus we set α and P_a as 1.0 and 0.25 respectively in the rest of the experiments.

In order to validate the importance of considering the limitation of the platform capacity, we conduct a control experiment which does not consider the platform capacity constraints. In the control experiment, we delete platform capacity constraints from the proposed model and adopt the CS algorithm to solve the new model. Comparing the solutions obtained by the new model and the original model, the total train delay time are 1672 s and 1526 s, respectively. As shown in Fig C2 and Fig C3, the total number of stranded passengers are 1263 pax and 1027 pax, respectively. It can be seen from the comparison of the results that the solution obtained by the new model is less effective than that obtained by the proposed optimization model. This is because ignoring the platform capacity leads to an increase in the number of stranded passengers on the platform and aggravates train delays at the same time.

In order to validate the effectiveness of the cuckoo search in solving automatic train regulation problems, we compared the proposed method with the standard PSO algorithm and a heuristic method called FRM proposed in paper [6]. The parameters for the PSO algorithm in this paper are same with it in study [3]. The timetable comparison results of the three methods mentioned above are shown in Fig C4, where the blue solid line represents the regulation solution by the cuckoo search algorithm, the green and red ones express the new schedule with the PSO and FRM algorithm, respectively. The calculation results for the total train delay with these three methods are 1526 s, 1597 s and 1901 s, respectively. As shown in Fig C4, the number of the affected trains is the least when using the cuckoo search algorithm and the regulation process has the shortest duration. The number of the affected trains is the same when using the PSO and FRM methods and the regulation process has the longest duration with the FRM algorithm. The calculation results of the total number of stranded passengers using the CS, PSO and FRM algorithm are 1027 pax, 1058 pax and 1413 pax, respectively. The distribution of the stranded passengers during the regulation process by using the three algorithms are illustrated respectively in Fig C2, Fig C5 and Fig C6. Compared with the FRM algorithm, the CS and PSO algorithm can efficiently reduce the number of passengers stranded on platforms and the number of stranded passengers by using

Table C3 Optimization results for different parameter sets

Test number	α, P_a	Optimal result	Worst result	Number of success	Number of runs	Success rate
1	0.5, 0.25	0.768	0.823	98	100	98%
2	0.5, 0.5	0.768	0.823	92	100	92%
3	0.5, 0.75	0.768	0.926	81	100	81%
4	1.0, 0.25	0.768	-	100	100	100%
5	1.0, 0.5	0.768	0.823	93	100	93%
6	1.0, 0.75	0.768	0.926	78	100	78%
7	1.5, 0.25	0.768	0.823	92	100	92%
8	1.5, 0.5	0.768	0.926	86	100	86%
9	1.5 0.75	0.768	0.926	74	100	74%

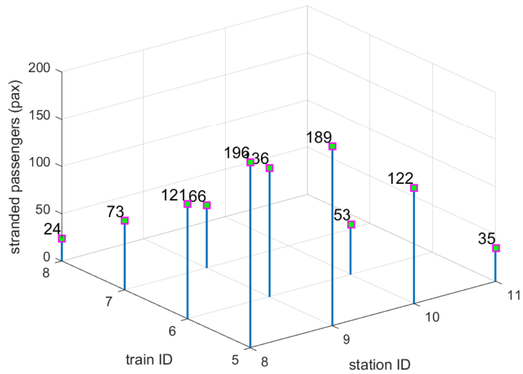

Figure C2 Total stranded passengers with the platform capacity constraints

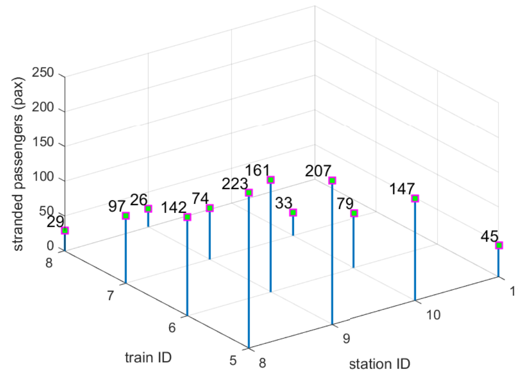

Figure C3 Total stranded passengers without the platform capacity constraints

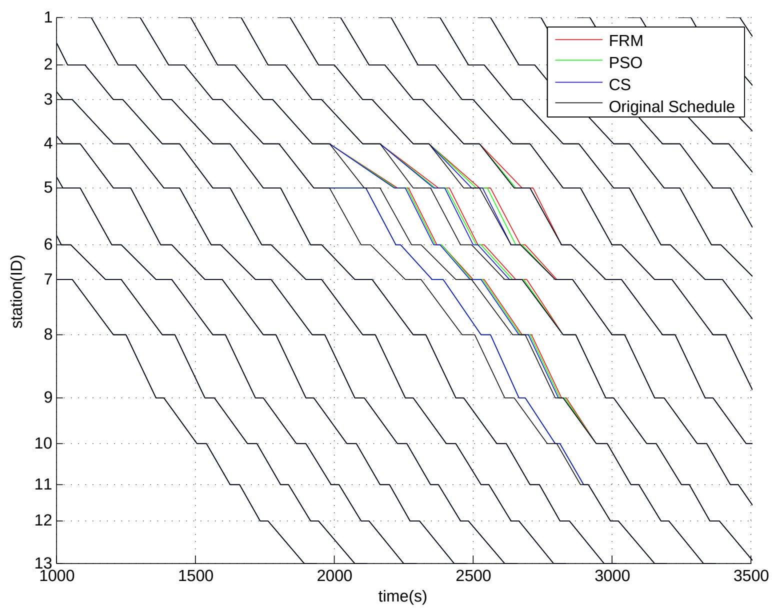

Figure C4 The comparison of three train diagrams

Table C4 Computation results of different delay scenarios

Delay scenario	Algorithm	The number of affected trains	Total train delay time (s)	The number of total stranded passengers (pax)	Optimal results	Computation time (s)
1. (4, 5, 100)	FRM	4	991	748	-	-
	CS	3	802	425	0.689	7.23
2. (6, 5, 100)	FRM	4	1029	912	-	-
	CS	3	831	671	0.772	7.31
3. (5, 8, 150)	FRM	6	2755	2036	-	-
	CS	5	2206	1719	0.823	7.56
4. (5, 8, 200)	FRM	8	5923	3183	-	-
	CS	6	4674	2859	0.844	7.61
5. (5, 8, 300)	FRM	11	16750	5505	-	-
	CS	8	12999	3975	0.749	8.05

the CS algorithm is the least.

The performance comparison between the CS and PSO algorithm is shown in Fig C7. The convergence curve of two objectives by using CS algorithm is illustrated in Fig C8 and Fig C9, respectively. The convergence speed and optimal solution of the CS algorithm outperform that of the PSO algorithm.

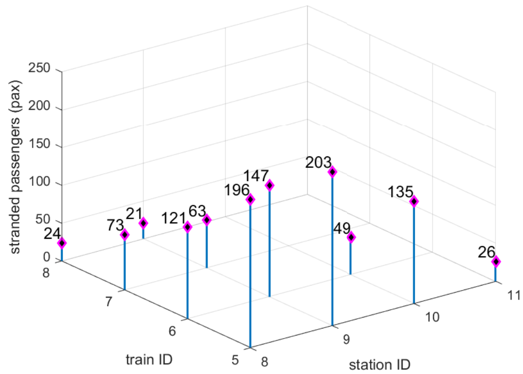

Figure C5 Total stranded passengers using the PSO

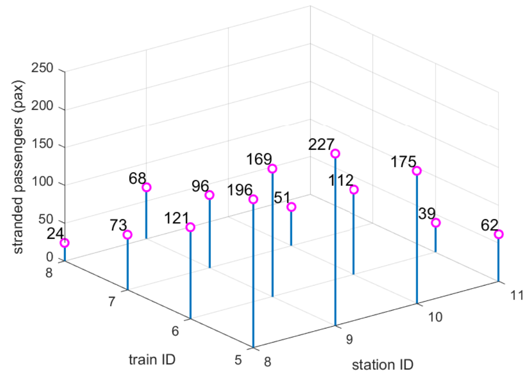

Figure C6 Total stranded passengers using the FRM

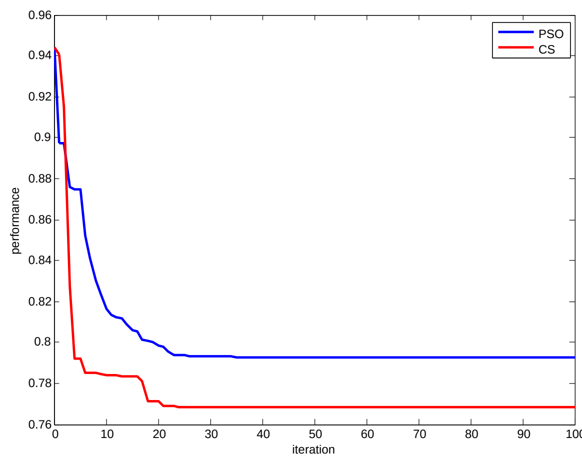

Figure C7 Convergence of the CS and PSO algorithm

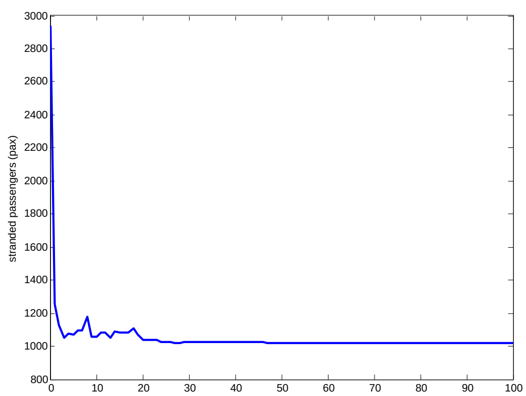

Figure C8 Convergence of the total stranded passengers of the CS

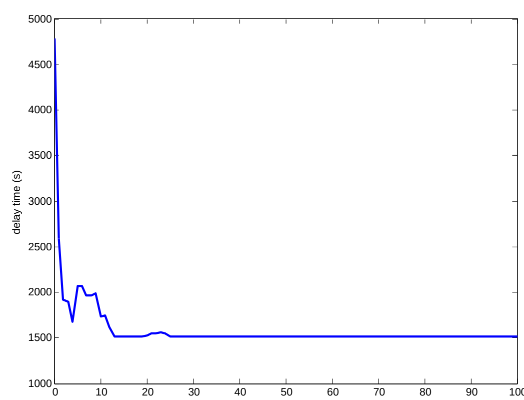

Figure C9 Convergence of the total delay of the CS

Appendix C.2 Different weights for objectives

In order to demonstrate the effect of different weight on the objective function, we consider more experiments in the following discussion. The delay scenario and the parameters of the cuckoo search algorithm are the same with that in Appendix C.2. The weight coefficients for the total train delay and the total number of stranded passengers on platforms is investigated, which the range is set as $a_1 \in (0.0, 1.0)$ and $a_2 \in (0.0, 1.0)$, and the sum of these two coefficients is equal to 1. The computational results of each weights set for the proposed model solving by the cuckoo search algorithm are listed in Table C5 and the corresponding performance variation curves are shown in Fig C10.

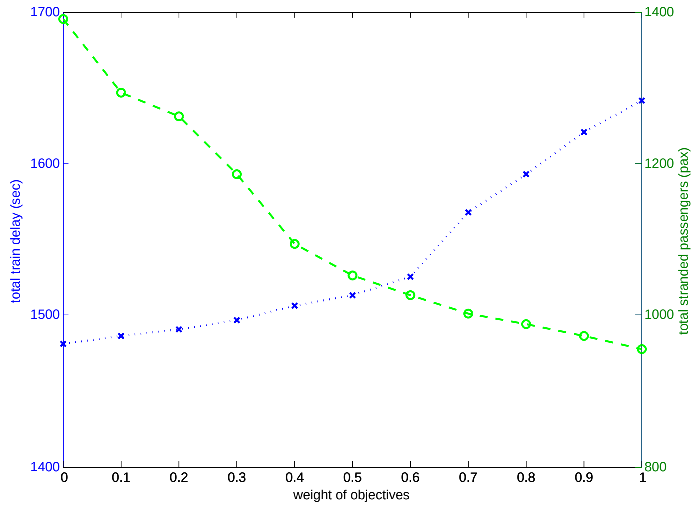

Figure C10 Different weights for objectives

Table C5 Computational results with different weights by the CS algorithm

a_1, a_2	Train delay (s)	Stranded passengers (pax)	Objective cost
0.0, 1.0	1695	963	0.682
0.1, 0.9	1647	972	0.706
0.2, 0.8	1631	981	0.727
0.3, 0.7	1593	994	0.744
0.4, 0.6	1547	1012	0.755
0.5, 0.5	1526	1027	0.765
0.6, 0.4	1513	1051	0.775
0.7, 0.3	1501	1136	0.794
0.8, 0.2	1494	1185	0.796
0.9, 0.1	1486	1241	0.791
1.0, 0.0	1478	1283	0.777

As shown in Fig C10, the value of each item will decrease as its weight increases. This can provide a clear guidance for metro operators, who can choose the appropriate set of coefficients to make a trade-off regulation decision between reducing the deviation from the original schedule and the total number of the left-behind passengers, or to achieve other purposes for the certain interests.

Appendix C.3 Different delay scenarios

In order to validate the reliability of the proposed method for solving automatic train regulation problem, we design several delay scenarios with different disturbance locations and the duration of initial delay. We use $(s, t, d_{s,t})$ to denote the delay scenario, where s represents the station where the disturbance occurs, t expresses the metro train in trouble and $d_{s,t}$ means the duration of the initial delay. The weights for each objective are set as 0.5 and 0.5, respectively.

The computation results for different delay scenarios are listed in Table C4. As the duration of the initial delay increases, the number of affected trains will increase and the corresponding total train delay time will also increase dramatically. This is because the reserved capacity of the original timetable is limited, longer initial delay will need more buffer times to be absorbed and thus more trains and passengers will inevitably be influenced. Due to the train and platform capacity limitation, the growth rate of the total number of stranded passengers is much slower than that of the total train delay time. In addition, the computation times of the proposed method for different scenarios (even for the larger disturbance that 8 trains need to be rescheduled) is short enough, which can satisfy the real-time requirements for automatic train regulation in practice.

Note that, for a certain metro line, the minimum allowable headway is determined. Thus, the larger the operated headway (i.e. the departure time interval) is, the higher the ability of recovery will be. However, a smaller departure time interval means the metro system can transport more passengers at the same control time window. If the operational headway becomes smaller (e.g., 150s or 120s), the metro system still can recover from delays since there is still the reserve headway capacity. However, the number of affected trains will be more than the condition when the departure time interval is 180s.

References

1 Sun L, Jin J, Lee D H, et al., "Demand-driven timetable design for metro services," *Transportation Research Part C*, vol. 46, pp. 284–299, 2014.

2 Niu H, Zhou X, "Optimizing urban rail timetable under time-dependent demand and oversaturated conditions," *Transportation Research Part C*, vol. 36, pp. 212–230, 2013.

3 Hou Z, Gao S, Nicholson G, et al., "A hybrid intelligent algorithm for real time metro traffic regulation in cases of disturbance," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, Nov 2018, pp. 3929–3934.

4 Li S, Yang L, Gao Z, et al., "Robust train regulation for metro lines with stochastic passenger arrival flow," *Information Sciences*, vol. 373, pp. 287–307, 2016.

5 Yin J, Tang T, Yang L, et al., "Energy-efficient metro train rescheduling with uncertain time-variant passenger demands: An approximate dynamic programming approach," *Transportation Research Part B Methodological*, vol. 91, pp. 178–210, 2016.

6 Hou Z, Dong H, Gao S, et al., "Energy-saving metro train timetable rescheduling model considering ato profiles and dynamic passenger flow," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 7, 2019.

7 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," *Transportation Research Part B: Methodological*, vol. 76, pp. 117 – 135, 2015.

8 Gao Y, Kroon L, Schmidt M, et al., "Rescheduling a metro line in an over-crowded situation after disruptions," *Transportation Research Part B Methodological*, vol. 93, pp. 425–449, 2016.

9 Li S, Dessouky M M, Yang L, et al., "Joint optimal train regulation and passenger flow control strategy for high-frequency metro lines," *Transportation Research Part B Methodological*, vol. 99, pp. 113–137, 2017.

10 Kang L, Wu J, Sun H, et al. , "A practical model for last train rescheduling with train delay in urban railway transit networks," *Omega*, vol. 50, pp. 29–42, 2015.

11 Kang L, Zhu X, Sun H, et al., "Last train timetabling optimization and bus bridging service management in urban railway transit networks," *Omega*, vol. 84, pp. 31 – 44, 2019.

12 Yang, X, "Nature-inspired metaheuristic algorithms second edition," 2010.

13 R. N. Mantegna, "Fast, accurate algorithm for numerical simulation of lévy stable stochastic processes." *Physical Review E Statistical Physics Plasmas Fluids Related Interdisciplinary Topics*, vol. 49, no. 5, p. 4677, 1994.