Paper:

A Mixed Encoding Genetic Algorithm for Train Platforming Rescheduling under Train Delays

Shuxin Ding[*1,*2], **Tao Zhang**[*1,*2,†], **Rongsheng Wang**[*1,*2,*3], **Yanhao Sun**[*1,*2], **Xiaozhao Zhou**[*1,*2], **Chen Chen**[*4,*5,†]

[*1]Signal and Communication Research Institute, China Academy of Railway Sciences Corporation Limited, Beijing 100081, China
[*2]Train Operation Control Laboratory for High-Speed Railway, National Engineering Research Center of System Technology for High-Speed Railway and Urban Rail Transit, Beijing 100081, China
[*3]Postgraduate Department, China Academy of Railway Sciences, Beijing 100081, China
[*4]School of Automation, Beijing Institute of Technology, Beijing 100081, China
[*5]Key Laboratory of Intelligent Control and Decision of Complex Systems, Beijing Institute of Technology, Beijing 100081, China
E-mails: 13701193534@139.com, xiaofan@bit.edu.cn
[†]Corresponding authors

To improve the operational efficiency of the high-speed railway station with emergencies, a train platforming rescheduling problem (TPRP) at a high-speed railway station is proposed in this paper. The TPRP problem addresses the adjustment of train track assignment and train arrival/departure time under train arrival delays. A mixed-integer nonlinear programming model is proposed to minimize the weighted sum of total train delays and rescheduling costs. A mixed encoding genetic algorithm (GA) is proposed. The individual of the GA is represented as the platform track assignment and train departure priority, which is a mixed encoding scheme with integer and permutation. The individual is decoded to a feasible platform track assignment and arrival/departure time of trains using a rule-based method for conflict resolution in platform tracks and arrival/departure route. Experimental results demonstrate the superiority of the mixed encoding GA in solving the proposed problem compared with other algorithms.

Keywords: high-speed railway, train platforming rescheduling, conflict resolution, genetic algorithm, mixed encoding

1. Introduction

The arrival/departure routes and platform tracks in the high-speed railway passenger station are important to the station transportation organization. It directly impacts the efficiency of station operations and the capacity of the station. When inevitable emergencies occur during train operations, e.g., infrastructure failure, bad weather, etc., train operations may be disturbed/disrupted with delays [3]. As a result, the original train platform schedule cannot meet the requirements of the station operations. Therefore, a fast and efficient adjustment of the train platform schedule, including the platform tracks assignment and arrival/departure time of the affected trains, should be done. So as to ensure the safety of train operation and recover their regular operation as soon as possible.

The problem of adjusting the train platform schedule is known in the literature as the train platforming rescheduling problem (TPRP) [8]. Various studies have been conducted on the TPRP problem, which has been proven to be NP-hard [8]. In most studies, the problem is formulated as a mixed-integer linear programming (MILP) model [1, 4, 13, 15] and mixed-integer nonlinear programming (MINLP) model [14]. Minimizing the total delay time of the trains is usually used as the optimization objective. Other objectives are minimizing the total platform track assignment costs [15], deviations from the original platform [8], etc. CPLEX solver is usually used as the tool for solving the TPRP problem. However, when the scale of the problem increases, the computation time of the CPLEX solver increases significantly.

Metaheuristics are usually used for solving NP-hard problems [2]. Zhang *et al.* [15] proposed a genetic and simulated annealing hybrid algorithm to solve the re-optimization of train platforming in case of train delays. Zhang *et al.* [14] proposed an improved discrete teaching and learning optimization algorithm to solve the problem. In most studies, only the train platform track assignment is used as the encoding for optimization. The arrival and departure times are adjusted through heuristics during the conflict resolution and are not directly controlled by the algorithm.

We summarize three contributions in this paper. First, the train platforming rescheduling problem with train delays is proposed and modeled as an MINLP problem. Second, an effective genetic algorithm is proposed, with a novel mixed encoding method with integer and permutation encoding schemes for solution representation and a rule-based decoding method to obtain a new train platform schedule. Finally, experimental results show the efficiency and effectiveness of the proposed GA compared with other algorithms.

The rest of this paper is organized as follows. The proposed model is formulated in Section 2. Section 3 presents a novel genetic algorithm for solving the TPRP. The performance of the proposed algorithm is evaluated in Section 4. Finally, conclusions and future work are provided in Section 5.

2. Problem Formulation

This section introduces an MINLP model to formulate the TPRP with train arrival delays. This model minimizes the weighted sum of total train delays and rescheduling costs.

2.1. Assumptions

There are seven assumptions. (1) We only consider the arrival and departure of trains at the station. Train passing through the station without stopping is not considered. (2) The settings of the train station, train platforming plan, and original train arrival and departure time are known. (3) The number of platform tracks and routes satisfies the need for train arrival and departure during regular operation. (4) The upstream and downstream trains are operated separately on their side of platform tracks and arrival and departure routes. We only consider the rescheduling on one side. (5) Disruption is not considered at the station, e.g., track blockage is not considered. (6) Some trains are delayed with known delay time. (7) The trains arrive from one direction, which means the order for occupying the routes and platform tracks (train arrival order) are determined according to the estimated arrival time.

2.2. Parameters and Decision Variables

For clarity, the notations of the proposed model are shown in **Table 1**.

2.3. Objective Function

The objective function in this model is a weighted sum of two parts. The first part Z_1 is the sum of the total train arrival and departure delays. The second part Z_2 is the rescheduling costs of the train platform schedule, including the total number of train arrival/departure time adjustments and train platform track adjustments. $\mathrm{sgn}(\cdot)$ returns 1 when the rescheduled arrival/departure time is later than the original arrival/departure time and returns 0 when the arrival/departure time remains the same. $\sum_{i\in I} 0.5|X_{l,i}-x_{l,i}|$ equals 1 when the train platform track is adjusted, and 0 otherwise.

$$Z_1 = \sum_{l\in L}(t_l^a - T_l^a) + \sum_{l\in L}(t_l^d - T_l^d) \tag{1}$$

$$Z_2 = \sum_{l\in L}\mathrm{sgn}(t_l^a - T_l^a) + \sum_{l\in L}\mathrm{sgn}(t_l^d - T_l^d) + \sum_{l\in L}\sum_{i\in I} 0.5|X_{l,i}-x_{l,i}| \tag{2}$$

Table 1. Summary of notations.

Symbol	Description
	Indices
i	index of platform track, $i \in I$
l,k	index of train, $l,k \in L$
	Parameters
I	the set of platform tracks
L	the set of trains
T_l^a	arrival time of train l in the orignal schedule
T_l^d	departure time of train l in the orignal schedule
T_S	safety interval time between two consecutive trains that occupy the same platform track
h^a	minimal headway between two consecutive arrival trains
h^d	minimal headway between two consecutive departure trains
w	weight value for rescheduling cost
$X_{l,i}$	platform track assignment of train l in the original schedule, 1 if train l occupies platform track i; 0 otherwise
d_l	arrival delay for train l
τ_l^a	estimated arrival time of train l
$q_{l,k}^a$	actual order for train arrival, 1 if train l arrives before train k; 0 otherwise
M	a large positive number
	Decision variables
$x_{l,i}$	actual platform track assignment of train l, 1 if train l occupies platform track i; 0 otherwise
t_l^a	actual arrival time of train l
t_l^d	actual departure time of train l
$q_{l,k}^d$	actual order for train departure, 1 if train l departures before train k; 0 otherwise

2.4. Constraints

Several constraints for train operations at the station are described as follows.

$$\sum_{i\in I} x_{l,i} = 1, \quad \forall l \in L \tag{3}$$

$$t_k^a - t_l^d \geq T_S q_{l,k}^a - M(3 - x_{l,i} - x_{k,i} - q_{l,k}^a), \quad \forall l,k \in L, l \neq k, i \in I \tag{4}$$

$$t_k^a - t_l^a \geq h^a q_{l,k}^a - M(1 - q_{l,k}^a), \quad \forall l,k \in L, l \neq k \tag{5}$$

$$t_k^d - t_l^d \geq h^d q_{l,k}^d - M(1 - q_{l,k}^d), \quad \forall l,k \in L, l \neq k \tag{6}$$

$$q_{l,k}^d + q_{k,l}^d = 1, \quad \forall l,k \in L, l \neq k \tag{7}$$

$$t_l^d - t_l^a \geq T_l^d - T_l^a, \quad \forall l \in L \tag{8}$$

$$\tau_l^a = T_l^a + d_l, \quad \forall l \in L \tag{9}$$

$$t_l^a \geq \tau_l^a, \quad \forall l \in L \tag{10}$$

$$t_l^d \geq T_l^d, \quad \forall l \in L \tag{11}$$

$$t_l^a, t_l^d \geq 0, \quad \forall l \in L \tag{12}$$

$$x_{l,i} \in \{0,1\}, \quad \forall l \in L, i \in I \tag{13}$$

$$q^{d}_{l,k} \in \{0,1\}, \quad \forall l,k \in L, l \neq k \tag{14}$$

where Eq. (3) guarantees that only one platform track is assigned to train l. Eq. (4) guarantees the safety interval time between two consecutive trains that occupy the same platform track. Eqs. (5) and (6) guarantee that the arrival and departure headways for any two trains satisfy the requirement for the station, which avoids conflicts in arrival and departure routes. Eq. (7) is the departure order constraint of two trains at the station. Eq. (8) guarantees the dwell time at the station for trains is greater or equal to the original one. Eq. (9) represents the calculation for estimated arrival time. Eqs. (10) and (11) guarantee the actual arrival and departure time is no less than the estimated arrival time and original departure time, respectively. Eqs. (12)–(14) restrict the decision variables to real and binary numbers.

2.5. Proposed Model

The TPRP model is formulated to minimize the weighted sum of the total train arrival/departure delays and the rescheduling costs of the train platform schedule under several constraints, that is:

$$\begin{cases} \min\ Z = Z_1 + wZ_2 \\ \text{s.t. Constraints } (3)-(14). \end{cases} \tag{15}$$

2.6. Model Reformulation

Due to the nonlinear terms (sgn$(\cdot)$ and $|\cdot|$) in Eq. (2), linearization method is developed. Three auxiliary variables are introduced, i.e., $\mathbf{r}_1 = [r^l_1]_{|L|\times 1}$, $\mathbf{r}_2 = [r^l_2]_{|L|\times 1}$, and $\mathbf{r}_3 = [r^{li}_3]_{|L|\times|I|}$, which are defined as follows:

$$\begin{cases} r^l_1 = \text{sgn}(t^a_l - T^a_l) \\ r^l_2 = \text{sgn}(t^d_l - T^d_l) \\ r^{li}_3 = |X_{l,i} - x_{l,i}| \end{cases} \tag{16}$$

Substituting Eq. (2) by Eq. (16), a reformulated MILP model is obtained. Eq. (2) is reformulated as

$$Z_3 = \sum_{l\in L} r^l_1 + \sum_{l\in L} r^l_2 + \sum_{l\in L}\sum_{i\in I} 0.5 r^{li}_3 \tag{17}$$

The TPRP model can be reformulated as follows:

$$\min\ Z = Z_1 + wZ_3 \tag{18}$$

$$\text{s.t. } Mr^l_1 \geq t^a_l - T^a_l, \quad \forall l \in L \tag{19}$$

$$Mr^l_2 \geq t^d_l - T^d_l, \quad \forall l \in L \tag{20}$$

$$r^l_1 \leq t^a_l - T^a_l, \quad \forall l \in L \tag{21}$$

$$r^l_2 \leq t^d_l - T^d_l, \quad \forall l \in L \tag{22}$$

$$r^{li}_3 \geq X_{l,i} - x_{l,i}, \quad \forall l \in L, i \in I \tag{23}$$

$$r^{li}_3 \geq x_{l,i} - X_{l,i}, \quad \forall l \in L, i \in I \tag{24}$$

$$r^l_1, r^l_2 \in \{0,1\}, \quad \forall l \in L \tag{25}$$

$$r^{li}_3 \in \{0,1\}, \quad \forall l \in L, i \in I \tag{26}$$

$$\text{Constraints } (3)-(14). \tag{27}$$

Eqs. (19)–(26) guarantee the reformulated model is equivalent to the original model. The reformulated model is a MILP model which belongs to NP-hard problems.

Algorithm 1 The novel genetic algorithm for TPRP

Input: Population size N_p.
Output: Final population P.

Generate the initial population P with N_p individuals randomly.
while terminate condition is not satisfied **do**
    Select parent individuals through roulette wheel selection.
    Update P through single-point crossover and modified order crossover.
    Update P through single-point mutation and swap mutation.
    Merge the new populations with the original ones and obtain the best individuals according to the population size.
end while
return

3. Proposed Method

This section proposes a novel genetic algorithm (GA) to solve the TPRP. First, encoding and decoding were introduced to transform the original MILP problem into an integer and permutation-based combinatorial optimization problem without constraints. The population of GA is updated by crossover and mutation operators. The GA process is shown in **Algorithm 1**.

3.1. Encoding and Decoding

For TPRP, most studies use the integer-value encoding scheme to represent the platform track assignment. When delays occur, affected trains are likely to change platform track which may face conflicts with other trains. We proposed a novel encoding for TPRP, which is a mixed encoding scheme with integer-value and permutation-value encodings. The integer-value encoding is similar to previous studies, which stands for the platform track assignment. The value range is within the range [1, $|I|$]. The length of the first part of an individual is equal to the number of trains $|L|$. The permutation-value encoding is used to determine the priority of the trains. When conflict occurs in determining the departure time of trains, the departure time of the train with lower priority is adjusted for conflict resolution. The value range is within the range [1, $|L|$]. The length of the second part of an individual is also equal to the number of trains $|L|$.

The key issue in decoding is how to minimize the change of train arrival/departure time to avoid causing delays and minimize the increase of the number of additional adjustments. Given an individual with platform track assignment and train departure priority, there are still three types of conflicts between trains at the station:

1) **Conflicts when trains occupy the same platform track.** This refers to Eq. (4). Since the actual order for train arrival is determined by the estimated arrival time. The arrival and departure time of the

subsequent train is adjusted according to the safety interval time constraint in Eq. (4).

2) **Conflicts in the arrival routes.** This refers to Eq. (5). Similar to the former conflict, since the order of the arrival trains is determined, the arrival and departure time of the subsequent train is adjusted according to the arrival headway constraint in Eq. (5).

3) **Conflicts in the departure routes.** This refers to Eq. (6). The actual order for the departure of affected trains is based on the train departure priority. The departure time of affected trains with lower priority is adjusted according to the departure headway constraint in Eq. (6).

Based on the above rules for conflict resolution, the constraints in the TPRP model can be effectively handled, and the model can be converted into an unconstrained one. The feasibility of the solution under mixed coding is guaranteed, and the efficiency in solving the original constrained optimization problem is improved significantly.

3.2. Population Initialization

The initial population is randomly generated. For the platform track assignment part, it is an integer randomly generated with the range [1, $|I|$]. For the train departure priority part, it is a permutation randomly generated with the range [1, $|L|$].

3.3. Selection Operator

The operator used for selection was the roulette wheel selection. It is typically used in GA. The individuals were selected according to their fitness values. Because this is a minimization problem, the individuals' probabilities are set according to the exponential of the negative fitness values.

3.4. Crossover Operator

According to the characteristics of the encoding, two crossover operators are adopted based on the crossover rate p_c, respectively [12].

1) Single-point crossover is adopted for the integer-value encoding. This operator selects two parents and then randomly selects a point for crossover. Two offspring are obtained by combining parents at a crossover point.

2) Modified order crossover (MOC) is adopted for the permutation-value encoding. The MOC operator randomly selects a crossover point to divide both parent individuals p_1 and p_2 left and right strings of the same length. Then, the order of the right string p_1 is used to change the order of the positions in p_2 and vice versa.

Table 2. Parameters settings for test instances.

Parameters	Values
$\lvert I\rvert$	6
$\lvert L\rvert$	60, 70, 79
T_S	3 min
h^a	4 min
h^d	4 min
w	1, 10
M	1440

3.5. Mutation Operator

According to the characteristics of the encoding, two mutation operators are adopted based on the mutation rate p_m, respectively.

1) Single-point mutation is adopted for the integer-value encoding. A position in an individual is randomly selected and replaced to obtain a new integer.

2) Swap mutation is adopted for the permutation-value encoding. Two positions in an individual are randomly selected and swapped to obtain a new permutation.

4. Computational Experiments

This section investigates the performance of the proposed GA. Test instances are first generated. Then, we solved the problem using the proposed GA and some algorithms for comparison, including exact solutions using CPLEX. All experiments were conducted on a PC with an Intel Xeon Gold 5218 CPU 2.30 GHz and 32 GB of internal memory. Exact solutions for TPRP problems were implemented in MATLAB R2020b using YALMIP as the modeling language and CPLEX 12.10, with default parameter settings [7]. Other algorithms for TPRP problems were implemented using MATLAB R2020b.

4.1. Test Instances

We first developed the test instances due to the lack of benchmark instances with train arrival delays for TPRP in the literature. Trains are running downstream at a high-speed railway station from 12:00 to 22:00 hrs. The arrival delay for train l (d_l) is a integer randomly generated within [1, 20] (min) at a occurrence probability of 0.5. Other parameters are shown in **Table 2**. Since the numbers of train sets in different instances are 60, 70, and 79, the numbers of trains with arrival delays are 29, 37, and 39. There are six test instances based on the combination of $|L|$ and w.

4.2. Algorithms for Comparison

To evaluate the performance of the proposed GA, we use the following three algorithms for comparison: Self-adaptive differential evolution algorithm (SaDE)

[11], comprehensive learning particle swarm optimizer (CLPSO) [6], and genetic algorithm (GA). The crossover and mutation operators are not the same as the proposed GA. It is a real-coded GA with arithmetic crossover [5] and Gaussian mutation [10]. These three algorithms are all searching in the continuous space. The integer value for platform track assignment is obtained by rounding, and the permutation value for train departure priority is obtained by the random key algorithm.

4.3. Parameter Settings

The population size N_p and the maximum number of generation *MaxGen* are set the same among algorithms, which are 200 and 1000. Therefore, the maximum number of fitness evaluations (*MaxFes*) is 2×10^5. For GAs, the crossover rate p_c and mutation rate p_m are set the same, which are 0.9 and 0.05. For CLPSO, the acceleration constant c is set as the original paper, which is 1.49445. The independent runs for each algorithm on each instance are set to 20.

4.4. Results and Analysis

We compare the performance of the proposed GA with three algorithms and CPLEX. **Table 3** shows the results of 20 independent runs for each algorithm, with mean values and standard deviations. The CPLEX runs only once. The best results of the metaheuristics are indicated in **bold**. It can be drawn from **Table 3** that the proposed GA outperforms the other three metaheuristics. The best values from the proposed GA are 944, 1032, 1132, 1775, 1995, and 2185, which approximate the exact value with a GAP of 1.17%, 1.07%, 1.15%, 2.08%, 2.61%, and 3.39%. As for the other three metaheuristics, since they are not designed for the mix encoding with integer-value and permutation-value, the performances are worse than the proposed GA. All the results of metaheuristics are obtained within 40 seconds, which guarantees real-time rescheduling.

Figs. 1 and **2** show the original and rescheduled train platform schedules for instance No. 2 obtained by the proposed GA with an objective value of 1032. In the figures, the black and blue rectangles are the occupation of different trains on the platform tracks and arrival/departure routes. The horizontal length of a rectangle is the occupation time on the tracks and routes. The number of the train is above the rectangle. The horizontal and vertical axes represent the time and name of the occupied track, respectively. The black rectangle represents the platform schedule of the train is not adjusted, while the blue rectangle represents the platform track or the arrival/departure time is adjusted.

For instance No. 2, the number of delayed arrival trains is 37. The number of total rescheduled trains is 52. The rescheduling cost is 115 (including 46 times of adjustments in arrival time, 49 times of adjustments in departure time, and 20 times of adjustments in platform track), and the total train delays are 917 min. It can be drawn from the **Figs. 1** and **2** that the unoccupied resources in the platform tracks (11G and 13G) are effectively used. For example, during 13:00 to 17:00 in track 13G and 15:00 to 18:00 in track 11G, tracks are unoccupied most of the time according to **Fig. 1**. While in the rescheduled train platform schedule, these tracks are more effectively used.

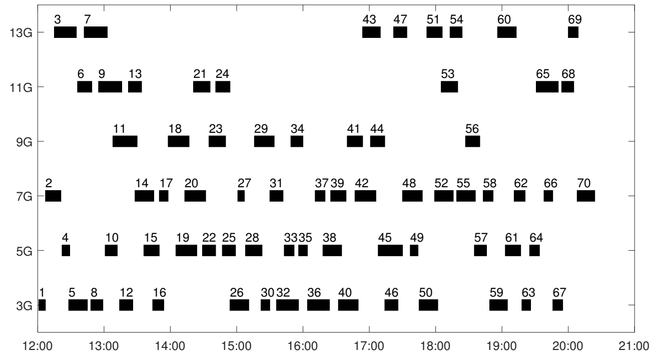

Fig. 1. Original train platform schedule for instance No. 2.

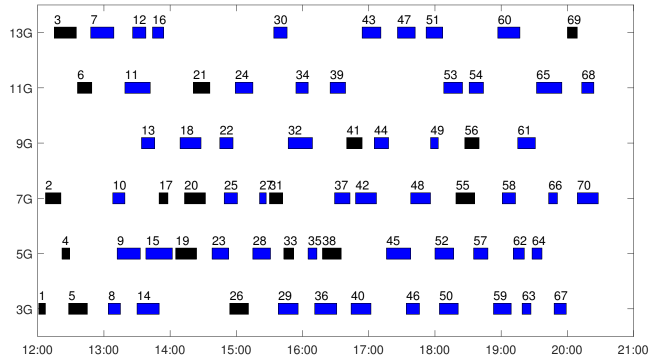

Fig. 2. Rescheduled train platform schedule for instance No. 2.

Figs. 3 and **4** show the convergence curves of the different algorithms in instances No. 3 and 6. The horizontal and vertical axes represent the number of fitness evaluations and the mean of the objective function for 20 runs, respectively. It can be observed from the figures that the proposed GA converges faster than the other algorithms at the beginning. In addition, both the proposed and real-coded GA have a high convergence speed. Finally, the final result of the proposed GA was better than those of the other algorithms.

5. Conclusion

The train platforming rescheduling at a high-speed railway station under train delays is formulated as an MINLP problem and linearized to a MILP model. A mix encoding GA is designed to solve TPRP. A novel encoding and decoding method is specially designed for the problem, transferring the original problem to an unconstrained one. This avoids a large amount of ineffective search in the solution space. After being tested in six test instances, the proposed GA outperforms other algorithms and shows its efficiency compared with CPLEX. The results can be obtained within 40 seconds, which is suitable for real-time

Table 3. Results of the comparison on the objective value of different algorithms.

Instance	$\|I\|/\|L\|/w$/delayed trains	SaDE	CLPSO	Real-coded GA	The proposed GA	CPLEX
1	6/60/1/29	991.95 ± 7.05	977.30 ± 5.21	963.80 ± 6.10	**950.20 ± 6.01**	933.00
2	6/70/1/37	1100.35 ± 7.42	1079.95 ± 5.60	1059.15 ± 4.72	**1042.25 ± 7.79**	1021.00
3	6/79/1/39	1242.75 ± 14.62	1212.15 ± 10.47	1173.40 ± 5.61	**1145.00 ± 6.94**	1119.00
4	6/60/10/29	2163.50 ± 16.16	2069.00 ± 24.89	1991.00 ± 37.63	**1863.75 ± 47.95**	1738.00
5	6/70/10/37	2425.95 ± 20.91	2342.40 ± 26.68	2233.80 ± 36.98	**2079.90 ± 48.76**	1943.00
6	6/79/10/39	2774.55 ± 38.01	2642.10 ± 27.86	2525.05 ± 55.71	**2267.95 ± 56.97**	2111.00

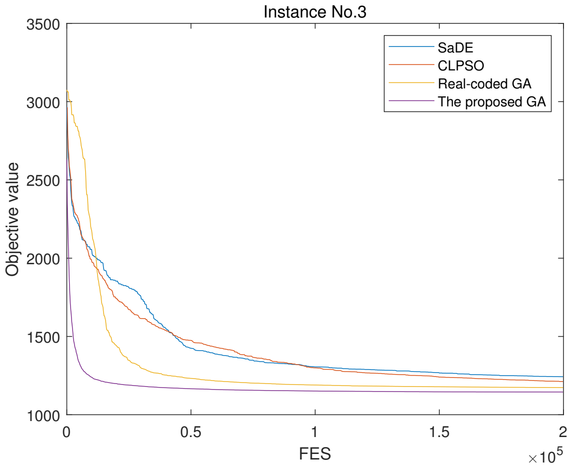

Fig. 3. Convergence curves of different algorithms for instance No. 3.

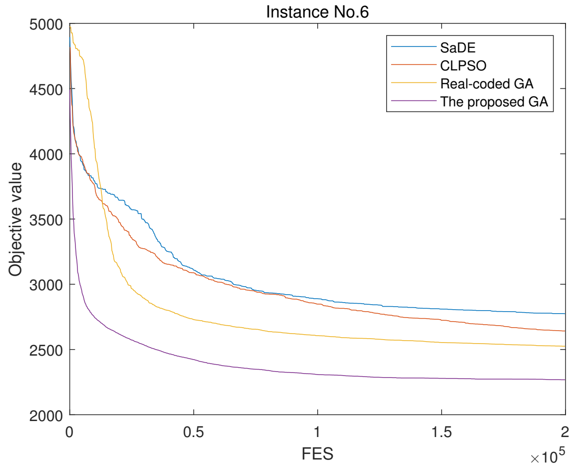

Fig. 4. Convergence curves of different algorithms for instance No. 6.

rescheduling.

In the future, we will consider a more complex railway station with more arrival/departure directions. Meanwhile, considering the uncertainties in the dynamic environment will make the model more practical. In addition, train timetable rescheduling and train platforming rescheduling can be further analyzed in an integrated model [9].

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grants U1834211, U1934220, 61790575, 62022015, in part by the Technological Research and Development Program of China Railway Corporation under Grant J2021G008, and in part by the Foundation of China Academy of Railway Sciences Corporation Limited under Grant 2021YJ315.

References:

[1] P. Chakroborty and D. Vikram, "Optimum assignment of trains to platforms under partial schedule compliance", Transportation Research Part B: Methodological, 42(2):169–184, 2008.

[2] S. Ding, C. Chen, Q. Zhang, B. Xin, and P. M. Pardalos, "Metaheuristics for resource deployment under uncertainty in complex systems", CRC Press, 2021.

[3] S. Ding, T. Zhang, Z. Liu, R. Wang, S. Lu, B. Xin, and Z. Yuan, "A memetic algorithm for high-speed railway train timetable rescheduling", Journal of Advanced Computational Intelligence and Intelligent Informatics, 26(3):407–417, 2022.

[4] R. García-Ródenas, M. L. López-García, L. Cadarso, and E. Codina, "A mixed-integer linear program for real-time train platforming management", Transportation Research Procedia, 62:815–823, 2022.

[5] P. Kora and P. Yadlapalli, "Crossover operators in genetic algorithms: A review", International Journal of Computer Applications, 162(10), 2017.

[6] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions", IEEE Transactions on Evolutionary Computation, 10(3):281–295, 2006.

[7] J. Löfberg, "YALMIP: A toolbox for modeling and optimization in MATLAB", In IEEE International Conference on Robotics and Automation, Taipei, Taiwan, 2004.

[8] G. Lu, J. Ning, X. Liu, and Y. M. Nie, "Train platforming and rescheduling with flexible interlocking mechanisms: An aggregate approach", Transportation Research Part E: Logistics and Transportation Review, 159:102622, 2022.

[9] X. Meng, Y. Wang, W. Xiang, and L. Jia, "An integrated model for train rescheduling and station track assignment", IET Intelligent Transport Systems, 15(1):17–30, 2021.

[10] S. Mirjalili, "Genetic algorithm", In Evolutionary algorithms and neural networks, pp. 43–55, Springer, 2019.

[11] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization", IEEE Transactions on Evolutionary Computation, 13(2):398–417, 2008.

[12] A. J. Umbarkar and P. D. Sheth, "Crossover operators in genetic algorithms: a review", ICTACT Journal on Soft Computing, 6(1):1083–1092, 2015.

[13] R. Wang, Y. Bao, and W. Hao, "Reallocating siding tracks in a railway station under severe disruptions", In 2018 International Conference on Intelligent Rail Transportation (ICIRT), pp. 1–5. IEEE, 2018.

[14] L. Zhang, W. Zheng, K. Huang, and H. Dong, "Research on optimization method of station track platform based on improved discrete teaching and learning optimization algorithm", In 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), pp. 3432–3437. IEEE, 2021.

[15] Y. Zhang, Q. Zhong, Y. Yin, X. Yan, and Q. Peng, "A fast approach for reoptimization of railway train platforming in case of train delays", Journal of Advanced Transportation, 2020, 2020.