A reinforcement learning based dynamic multi-objective constrained evolutionary algorithm for open-pit mine truck scheduling

Junxiang Qiu
School of Automation, China University of Geosciences
Hubei Key Laboratory of Advanced Control and Intelligent Automation for Complex Systems
Engineering Research Center of Intelligent Technology for Geo-Exploration, Ministry of Education
Wuhan 430074, China
15271934577@163.com

Changhe Li
School of Automation, China University of Geosciences
Hubei Key Laboratory of Advanced Control and Intelligent Automation for Complex Systems
Engineering Research Center of Intelligent Technology for Geo-Exploration, Ministry of Education
Wuhan 430074, China
changhe.lw@gmail.com

Shengxiang Yang
School of Computer Science and Informatics
De Montfort University
Leicester, UK
syang@dmu.ac.uk

***Abstract*—Aiming at the truck scheduling problem in the open-pit mine scenario, a truck scheduling model based on real-time ore blending is established, and an adaptive evolution algorithm for truck scheduling based on DCNSGA-III is proposed. In the established scheduling model, the real-time grade variance of the crushing plant is minimized as one of the optimization objectives, and the Q-learning algorithm is introduced to adaptively select one of the most effective operators during the search process. Experiments show that the proposed method can effectively control the grade fluctuation of the ore flow and better scheduling schemes are obtained in comparison with algorithms equipped with the traditional search operator selection methods.**

***Keywords*—truck scheduling, real-time ore blending, search operator selection, DCNSGA-III**

I. Introduction

Open-pit mining utilizes mining facilities in the mining area to excavate and extract rock ore and relies on the transportation system to convey the mined rock and ore to the crushing station and dumping site [1]. The order of transportation needs to consider real-time ore blending. An unreasonable transportation sequence will lead to large fluctuations in the grade of ore flow in the crushing station, which will make the subsequent beneficiation work more difficult, resulting in large fluctuations in the grade and recovery rate of finished ore. Meanwhile, As a crucial component of the open-pit mine production system, the truck transportation system holds a significant position in open-pit mining, with its investment accounting for approximately 40% to 60% of the total investment of the open-pit mine production system. Moreover, the transportation cost represents about 30% to 40% of the total mine production cost [2]. By implementing a reasonable and effective truck scheduling scheme, it becomes possible to enhance the utilization rate of truck equipment, reduce queuing waiting time, ensure a stable ore flow grade transported to the concentrator plant, and optimize the utilization of energy and mineral resources. Consequently, this leads to cost savings in dispatching and transportation, increased mine output, and improved economic benefits.

Establishing a reasonable open-pit mine truck scheduling model is essential for solving the truck scheduling problem. This model should be tailored to the actual production situation and take into account various influencing factors from different perspectives. For instance, literature [3] proposes a single-objective scheduling model aimed at minimizing fuel consumption to reduce energy usage in open-pit mine trucks. However, in practical production processes, truck scheduling often requires consideration of multiple objectives, as a single target fails to capture the multi-objective nature of the problem. To address this, literature [4] establishes a multi-objective scheduling optimization model based on truck queuing waiting time and trucking distance, while literature [5] constructs a scheduling model with objectives such as minimizing transportation cost, reducing total queuing waiting time, and minimizing the average grade deviation rate of the crushing station. Nevertheless, merely minimizing the average grade deviation rate might not ensure the stability of real-time ore flow and cannot meet the production demand.

Designing an excellent optimization algorithm to solve the truck scheduling problem is the second objective. In the past, as seen in the literature [6], the integer programming method was employed to solve the integer linear programming (MIP) model for truck scheduling. However, with the significant advancement of intelligent algorithms, intelligent optimization algorithms have emerged as the preferred mainstream approach. For instance, in literature [7], the NSGA-II algorithmic framework was utilized to address a multi-objective truck scheduling model by incorporating crossover operators and local search operators based on Variable Neighborhood Search (VNS) to generate offspring. Similarly, literature [8] introduced a hybrid heuristic algorithm called GGVNS, which combines the Greedy Randomized Adaptive Search Piecewise (GRASP) algorithm and Generalized Variable Neighborhood Search (GVNS) to optimize truck schedules. In the actual production process, one key component of an optimization algorithm is the search operator. The quality of optimization largely depends on the

This work was supported in part by the National Natural Science Foundation of China under Grants 62076226, in part by the Fundamental Research Funds for the Central Universities China University of Geosciences (Wuhan) under Grant CUGGC02, in part by the 111 project under Grant B17040.

∗ Corresponding author: Changhe Li (changhe.lw@gmail.com).

rationality of the search operator and the selection of appropriate search operators.

This paper presents a multi-objective scheduling model for open-pit mine trucks that takes account of real-time ore blending. One of the optimization goals is to minimize the real-time grade variance of ore flow in ore crushing stations. The optimization algorithm's framework is based on the dynamic constraint nondominated sorting genetic algorithm III (DCNSGA-III), and it utilizes multiple search operators tailored to the scheduling model's objectives and constraints. Additionally, an adaptive selection mechanism for search operators is designed using the Q-learning algorithm. The algorithm is tested using actual data from a mine, and the results demonstrate that the proposed algorithm more accurately identifies excellent and feasible scheduling schemes than traditional search mechanisms.

II. Truck Scheduling Model

The problem of open-pit mine truck scheduling involves efficiently scheduling and distributing multiple types of trucks to transport ore with varying grades from the loading point to the unloading point in the open-pit mine. To achieve this, various factors are considered, such as the type and quantity of truck equipment, the number of loading and unloading points, the ore grade sequence, ore quality requirements, transportation road network, and driving time with fuel consumption. The goal is to minimize waiting times and reduce ore flow grade fluctuations at the ore crushing plant while fulfilling the ore production and transportation demand. The real-world problem is transformed into a mathematical model, leading to the establishment of a constrained multi-objective truck scheduling model with four objectives and five constraints.

A. *Variable definitions*

Table I defines the parameters and variables used in this article.

TABLE I. Model Parameters

Parameters	Meaning
r	Truck number
i	Loading point number
j	Unloading point number
C_r	The load capacity of truck r
o_{rij}	The fuel consumption of truck r from i to j
x_{rij}	The number of heavy loads of truck r from i to j
y_{rij}	The number of no-load truck r from j to i
e	Group number of trucks arriving at the ore-crushing plant
v	The number of the first truck in the group
w	Number of trucks in the group
α_u	Ore grade of truck u
α_j	Ore grade indicator of unloading point j
t_r	Scheduling scheme execution time of truck u
t_0	Shift time or split time
g_i	Reserves of loading point i
p_j	Yield requirements of unloading point j
γ_j	Matching requirements of unloading point j
β_j	Kaolin content requirements of unloading point j
η_j	Hardness requirements of unloading point j

B. *Model building*

In the open-pit mining scenario, the key considerations are increasing the economic benefits of the mine and reducing production costs. Thus, two optimization goals are set to maximize the total ore output and minimize the total fuel consumption of the trucks, as represented in Equation (1) and Equation (2), respectively. Moreover, truck transportation significantly influences the subsequent beneficiation process, and maintaining a stable ore flow to the ore-crushing plant can greatly reduce beneficiation complexities. Therefore, the third optimization objective is to minimize the real-time grade variance of the ore flow in the ore crushing plant, as shown in Equation (3). In this equation, the trucks entering the ore crushing plant are grouped using a sliding window based on their entry time, enabling the calculation of grade fluctuation deviation within each window period, ensuring minimal ore grade fluctuations during each window. The fourth optimization goal is established based on the "divide and conquer" concept applied in the subsequent algorithm design. The entire working period is divided into multiple periods for optimization and solving, necessitating the scheduling scheme of each truck in each period to be as close as possible to the split time. This implies minimizing the variance of the working time of the truck, as shown in Equation (4).

$$F_1(s) = \max \sum_{r=1}^{k} \sum_{i=1}^{n} \sum_{j=1}^{m} C_r x_{rij} \ , \tag{1}$$

$$F_2(s) = \min \sum_{r=1}^{k} \left(\sum_{i=1}^{n} \sum_{j=1}^{m} o_{rij} x_{rij} + \sum_{i=1}^{n} \sum_{j=1}^{m} o_{rji} y_{rji} \right), \tag{2}$$

$$F_3(s) = \min \frac{1}{f} \sum_{e=1}^{f} \left| \frac{\sum_{u=v}^{v+w} C_u \alpha_u}{\sum_{u=v}^{v+w} C_u} - \alpha_j \% \right|^2 , \tag{3}$$

$$F_4(s) = \min \frac{1}{k} \sum_{r=1}^{k} \left| t_r - t_0 \right|^2 , \tag{4}$$

subject to

$$\sum_{r=1}^{k} \sum_{j=1}^{m} C_r x_{rij} - g_i \le 0 , \tag{5}$$

$$\sum_{r=1}^{k} \sum_{i=1}^{n} C_r x_{rij} - p_j \ge 0 , \tag{6}$$

$$\left| \frac{\sum_{u=v}^{v+w} C_u \gamma_u}{\sum_{u=v}^{v+w} C_u} - \gamma_j \% \right| \le 0 , \tag{7}$$

$$\left| \frac{\sum_{u=v}^{v+w} C_u \beta_u}{\sum_{u=v}^{v+w} C_u} - \beta_j \% \right| \le 0 , \tag{8}$$

$$\left| \frac{\sum_{u=v}^{v+w} C_u \eta_u}{\sum_{u=v}^{v+w} C_u} - \eta_j \% \right| \le 0 , \tag{9}$$

where constraint (5) indicates that the amount of ore mined at each loading point cannot exceed its ore reserves. Constraint (6) indicates that the total amount of ore discharged at each unloading point must be greater than or equal to its production requirement. Constraints (7), (8), and (9) define the ore quality requirements for each unloading point. They specify the desired ratios of powder ore and lump ore, kaolin content, and ore hardness requirements, respectively. The calculation method is similar to Equation (3), ensuring that the ore quality meets the specified requirements within each window period.

III. TRUCK SCHEDULING METHOD

The open-pit mine truck scheduling problem is a complex multi-constraint multi-objective optimization problem. To address the constraint optimization aspect of the problem, the dynamic constraint multi-objective optimization algorithm framework (DC) [9] is employed. Additionally, to handle the multi-objective optimization aspect, the NSGA-III algorithm framework [10] is utilized. The solutions to the problem are encoded using a character representation, where uppercase and lowercase letters correspond to loading and unloading points, respectively. Consequently, a truck's scheduling scheme can be represented by a string composed of alternating uppercase and lowercase characters.

The algorithm framework of the open-pit mine truck scheduling optimization algorithm based on the DCNSGA-III algorithm is shown in Algorithm 1.

Algorithm 1: DCNSGA-III

Input: population size N, termination T;
Output: Pareto optimal solution;
Step 1: initialization;
Generate initial population P_0;
Set constraint boundaries ε=$\varepsilon^{(0)}$;
Set t=0, s=0 , t_s=0;
Step 2: shrink ε;
if population is ε-feasible **then**
    $\varepsilon^{(s)}$=$\varepsilon^{(s+1)}$;
    Update population ε-feasible;
    s=s+1;
else
    t_s=t_s+1;
end if
Step 3: generate offspring;
Step 4: environmental selection is made using the NSGA-III algorithm;
Step 5: t=t+1;
Step 6: if the termination condition is met, go to step 7, otherwise, go to step 2;
Step 7: Output the Pareto optimal solution.

A. Initialization

With the principle of minimizing queuing waiting time, all trucks are alternately assigned loading and unloading points until the execution time of all truck scheduling scenarios reaches the specified time or all loading points are exhausted. This approach aims to achieve a feasible solution while minimizing waiting times for trucks as much as possible.

B. Evaluation of the solution

Before the calculation of objective values, constraint (5) is first checked. If it does not hold, the objective values of the solution are directly assigned the largest or minimal domain value to make this solution have the largest probability of being removed during the evolution process. Otherwise, perform a regular calculation.

C. Generation of offspring

We first find feasible solutions from the population and then select one of the following operators to generate offspring. We design ten search operators, i.e., LS1 to LS10, which aim to improve the solution from different situations.

LS1: Randomly select a truck and randomly swap two scheduling tasks in its scheduling scheme.

LS2: Select any truck and randomly reverse the order of the two scheduling tasks in its scheduling scheme.

LS3: Use the queue wait time heuristic to find the loading or unloading point with the longest queuing time among all trucks and replace its loading or unloading point.

LS4: Find out which scheduling scheme has the largest deviation between the work time and the set time, and replace its last loading or unloading point.

LS5: Find out the scheduling scheme in which the work time of the scheduling scheme deviates the most from the set time, and add scheduling tasks to it.

LS6: Find out the truck with the largest deviation in the ore grade deviation in the truck unloading sequence of the ore crushing plant and replace the previous loading point in its scheduling scheme.

LS7: Select any truck, select one loading point in its scheduling scheme, and replace it with a different loading point.

LS8: Select two trucks arbitrarily, and then select any scheduling task in the scheduling scheme of each vehicle to exchange.

LS9: Find out the two scheduling schemes with the longest work time and the shortest work time, and stitch the last scheduled task of the scheduling scheme with the longest work time to the last scheduling scheme with the shortest work time.

LS10: Find out the scheduled tasks in all scheduling schemes that exceed the set time and delete them.

In the above-mentioned search operators, a scheduling task refers to a combination of one loading point and one unloading point. The operators are classified as follows: LS1 to LS7 are designed to adjust one scheduling scheme, LS8 to LS9 are designed to adjust two scheduling schemes simultaneously, and LS10 is designed to modify all scheduling schemes simultaneously. This classification primarily facilitates later comparative experiments. Looking at the design principles of each search operator from another perspective: LS1, LS2, and LS8 are cross operators; LS7 is the variation operator; LS3 aims to reduce the waiting time of trucks; LS4, LS5, LS9, and LS10 aim to ensure that the work time of each truck's scheduling scheme falls within a certain range; and LS6 aims to reduce the grade fluctuation of ore crushing plants.

In the process of open-pit mining, the loading point may alternate shoveling of ore (grade greater than 0.3) and rock (grade less than 0.3), and the corresponding unloading points for ore and rock are different. Consequently, incorrect unloading points can easily occur in the subsequent generation process, which violates the production process. To address this issue, this paper adopts a repair method, starting from scratch to identify and fix the error in unloading points. To minimize restoration costs, the paper employs the idea of "divide and conquer," breaking down the scheduling scheme of the entire shift into multiple periods for resolution. Each period is solved based on the results from the previous one.

D. Adaptive selection of searching operators

Different search operators exhibit varying capabilities for improving individuals in different states. During the population evolution process, to adaptively select search operators with the most successful operator for specific individual states. The paper employs a reinforcement learning approach, utilizing Q-learning to enable the adaptive selection of search operators. This adaptive process enables the algorithm to prioritize the most effective search operators for each individual state, enhancing the overall efficiency and effectiveness of the population evolution process.

1) The definition of state space

The size of the state space in this paper is $4 \times J1 \times J2$, and each individual is described by 5 state characteristics. The first four features are the objective function value characteristics of individuals, and the last feature is the diversity characteristics of individuals. Each state feature is calculated by normalizing and then taking the remainder of $1/J1$ or $1/J2$. The normalized value is calculated as shown in Equations (10) and (11).

$$s_i = \frac{F_i - F_{i\min}^{'}}{F_{i\max}^{'} - F_{i\min}^{'}} \quad (i = 1,2,3,4), \tag{10}$$

$$s_5 = \frac{diff}{diff'}, \tag{11}$$

$$value = \frac{1}{m}\sum_{j=1}^{m}\sum_{i=1}^{i=4} a_i \frac{F_i - F_{i\min}^{'}}{F_{i\max}^{'} - F_{i\min}^{'}}, \tag{12}$$

$$diff = \sum_{j=1}^{m}\left(\sum_{i=1}^{i=4} a_i \frac{F_i - F_{i\min}^{'}}{F_{i\max}^{'} - F_{i\min}^{'}} - value\right), \tag{13}$$

where F_i represents the value of the i-th objective function of an individual, $F_{i\min}^{'}$ and $F_{i\max}^{'}$ represent the minimum and maximum values of the i-th objective function in the first generation population, *m* represents the number of individuals in the population, a_i represents the weighted value of the i-th objective function value, *value* represents the average weighted fitness value of the population, *diff* represents the diversity of the population, and $diff'$ represents the diversity of the first generation population.

2) The definition of action space

The main purpose of utilizing the Q-learning algorithm in this paper is to select a search operator with a strong improvement ability based on an individual's state. The action space for the Q-learning algorithm consists of the choices of different search operators, as shown in Equation (14).

$$A = \{LS1, LS2, \ldots, LS10\} \tag{14}$$

3) Reward function

In this paper, the scheduling goal is to maximize or minimize a specific indicator, and the reward function is designed to reflect the ability of an action to improve an individual. Considering different stages of evolution, the reward function consists of two parts. The first part involves the weighted sum of differences between the first four state characteristics of the child (next state) and the parent (current state). If a difference is negative, it is set to zero, and vice versa. If the offspring dominates the parent, then the reward in the first part is assigned a higher value. The second part is based on the number of objective functions in which the offspring performs better than the parent. This is taken into consideration because in the later stages of evolution, the population tends to converge, and the difference in objective function values becomes smaller compared to earlier stages. Relying solely on the difference in objective function values may not adequately express the improvement ability of search operators. The final reward is determined by combining the weights of the two parts. The weights of the first part decrease as the evolutionary process progresses, while the weights of the second part increase correspondingly, as shown in Equation (15).

$$reward = \beta_1 \sum_{i=1}^{4} a_i \left(s_i^{'} - s_i\right) + \beta_2 num \tag{15}$$

where a_i represents the weight of the i-th objective function, the same as in Equation (12), $s_i^{'}$ represents the i-th state feature of the child (the next state), s_i represents the i-th state feature of the parent (current state), *num* represents the number of objective functions in which the child is better than the parent. β_1 and β_2 represent the weight of the first and second parts of the reward.

4) The selection strategy of the action

The commonly used action selection strategy for the Q-learning algorithm is the ε-greedy strategy [11]. This strategy allows the algorithm to choose the optimal action with a high probability for generating offspring. However, in the early stages of evolution, the evolutionary algorithm needs to explore sufficiently to prevent the population from prematurely converging to a specific local extreme solution. On the other hand, it should converge quickly in the later stages of evolution. To achieve this balance, this paper enhances the ε-greedy strategy, making the probability of choosing the optimal action increase with the number of evolutionary iterations, as shown in Equation (16). This modification aims to encourage exploration in the early stages and gradually favor exploitation in the later stages of the evolutionary process.

$$\begin{cases} \varepsilon > \varepsilon_0 + (1-\varepsilon_0) * e^{-T/T_{\max}} & \text{choose optimal action} \\ \\ otherwise & \text{randomly choose action} \end{cases} \tag{16}$$

where ε is a random number, ε_0 represents the initial probability, T represents the current number of iterations and $T_{\max}$ is the maximum number of iterations.

Algorithm 2: generate offspring
Input: feasible individual *P*, the current number of iterations *T*, period *N*;
Output: offspring *Q*;
1 **Step 1:** calculate the individual *P* status;
2 **Step 2:** select the search operator;
3 **if** satisfy expression (16) or $N>N_0$ **then**
4 Select the search operator according to the Q table;
5 **else**

Select the search operator according to the random rule;
end if
Step 3: generate offspring Q based on the search operator;
Step 4: Calculate individual Q status and reward, and update the Q table.

IV. Experimental results and analysis

A. Experimental data

In this paper, an open-pit copper mine serves as the experimental scenario, and the actual production data of this mine is utilized as the experimental data. The production data comprises three main components: mine road network data, loading point data, and equipment data. The mine road network includes seven loading points (A-G), three unloading points (a-c), and 13 road network nodes. Among the loading points, A-E are the current shovel working points. The partial distance data from the loading points to the unloading points are presented in Table II. The loading point data primarily includes the ore output sequence of each loading point, with ore grades ranging from 0.70 to 0.87. As an example, the output sequence for loading point A is provided in Table III, containing information such as ore amount, ore grade, and ore hardness. The equipment used in the experiment consists of 15 220-ton trucks and 5 25-cubic meter forklifts, with one forklift assigned to each loading point. Specific parameters for the equipment can be found in Table IV.

TABLE II. Road network data

Loading point	Unloading point	Distance(m)
A	a	6343.03
B	a	3257.16
C	a	1324.58

TABLE III. The sequence of produced ore at loading point A

Output serial number	Amount of ore(t)	Ore grade	Ore hardness	Powder mineral content	Kaolin content
1	5725.25	0	0.28	0.11	0.14
2	7988.62	0.86	0.23	0.10	0.10
3	7899.86	0	0.27	0.12	0.11
4	177.52	0.74	0.30	0.13	0.19

TABLE IV. Underlying data

Parameter	Value
Loading time	3.8min
Unloading time	1.5min
Vehicle speed	22km/h
Shift time	10h

TABLE V. Parameter settings

Parameter	Value
a_i	0.25
ε_0	0.6
$J1$	4
$J2$	3
α_j	0.72
γ_j、β_j、η_j	1

B. Experimental results

The experiment is conducted on a system equipped with a core-i9 2.8GHz processor and 16GB RAM. All algorithms are implemented in C++. After several tests, the relevant parameters of the algorithm are set to the values specified in Table V. Following the idea of divide and conquer, we divide the entire 10-hour shift into 10 sections to address the problem.

1) Performance of Optimization Methods

To visually evaluate the algorithm's performance, we compare the best solution of the initial population (waiting time minimization) with the best solution of the final population (algorithm). The best solution is defined as the one with the minimum weighted sum of all normalized target values. Table VI demonstrates that the algorithm effectively reduces the real-time grade fluctuation of the ore flow in the ore crushing station, increases ore production and energy efficiency (ore production per unit of energy consumption), and reduces the variance of the work time for the scheduling scheme.

TABLE VI. Results of the initial and final solutions

Best solution	Ore production (t)	Energy efficiency (t/L)	Grade variance (%^{2})	Work time variance (min^2)
Initial	9680	3.87	0.00076	384.86
Final	10340	5.02	0.00038	27.76
Improvement ratio	6.81%	29.7%	50.0%	92.7%

2) Adaptive mechanism performance comparison

To verify the performance of the adaptive mechanism in the algorithm in this paper, firstly, we give the graphs of the selection probability of the search operators and the graphs of the improvement rate of each target of the best individual concerning the initial best individual for 9 consecutive periods, as shown in Figures 1 and 2, respectively. Finally, this paper compares four versions of the algorithm: the original algorithm, the improved algorithm based on rules (waiting time minimization), the improved algorithm based on the adaptive neighborhood local search mechanism (ANLS) [12], and our improved algorithm based on Q-learning algorithm, which is compared from the optimization results of the entire shift, as shown in Table VII.

Figure 1 gives the search operator selection probabilities for 9 consecutive periods, the first 6 time periods being the online learning phase, and the last 3 time periods selecting the optimal search operator directly based on the Q-table and individual states. From the first 6 time periods, at the beginning of evolution, the selection probability of each search operator does not differ much; however, as evolution proceeds, the algorithm adaptively selects the optimal search operator according to the individual state with a greater probability to make the population converge, and the selection probability of certain search operators becomes relatively higher. From the latter 3 time periods, the population can converge quickly because the optimal search operator is directly selected for evolution.

Figure 2 gives the improvement rate of each objective for the best individual for 9 consecutive periods. Combined with Figure 1, it is obvious from the (h) of Figure 1 and the (h) of Figure 2 that the improvement rate of the working time variance is lower after convergence is reached, and at this time the probability of selecting the LS4 that reduces the working time variance rises, and consequently the improvement rate of the best individual in the goal of the working time variance in also shows an upward trend. Similarly, this can be seen from the (i) of Figure 1 and the (i) of Figure 2. It shows that the algorithm proposed in this paper can realize the adaptive selection of search operator better to some extent and the designed search operator is effective.

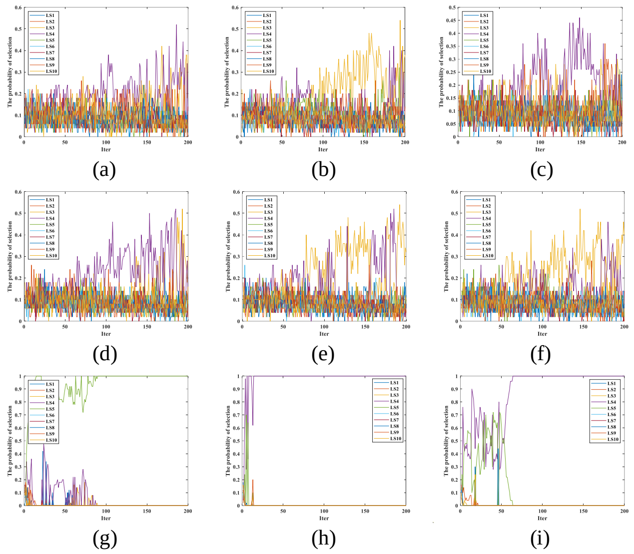

Fig. 1. Search operator selection probability graphs (a-i corresponding period 1-9)

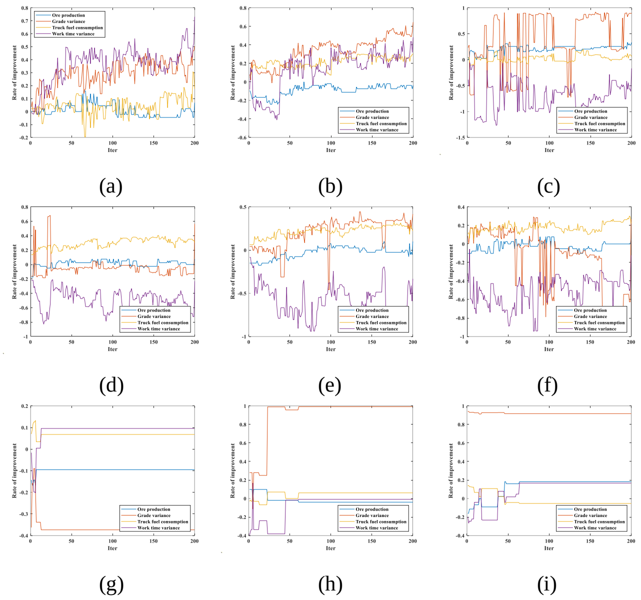

Fig. 2. Target improvement rate graphs (a-i corresponding period 1-9)

Table VII displays the results of the comparison of the final scheduling scenarios for the entire shift. From the data, it is evident that the algorithm utilizing Q-learning outperforms the other three versions concerning ore production and grade deviation, while all four versions of the algorithm are comparable in terms of the standard deviation of working time and the indicator of the amount of ore transported per unit of fuel consumption. Moreover from the Hyper volume metric and set coverage metric, the solution set of the algorithm proposed in this paper is better than the other three versions of the algorithm.

TABLE VII. THE FINAL SOLUTION FOR THE ENTIRE SHIFT GENERATED BY FOUR ALGORITHMS (ADD DIFFERENT MECHANISMS)

Final solution	**Ore produc-tion(t)**	**Grade varian-ce($\%^2$)**	**Work time varian-ce(min^2)**	**Hyper volume**	**set coverage**
Q-learning	115280	0.0021	45.50	0.18	standard
None	112200	0.0022	34.12	0.13	94%/0%
rules	114400	0.0023	54.07	0.00	100%/0%
ANLS	113080	0.0025	50.54	0.14	32%/8%

V. CONCLUSION

This paper comprehensively considers the production demand of open-pit mines and incorporates real-time ore blending into the modeling process. A truck scheduling model is established to minimize the real-time grade variance of ore flow in the ore crushing station as one of the optimization objectives. Additionally, based on the DCNSGA-III algorithm, the Q-learning algorithm is introduced to achieve an adaptive selection of search operators and solve the truck scheduling model. The experimental results show that the proposed method effectively controls the real-time grade fluctuation of ore flow in the ore crushing station, demonstrating superior performance compared to traditional search operator selection algorithms.

REFERENCES

[1] G. Wang, H. Wang, H. Ren, G. Zhao, Y. Pang, Y. Du, et al, “2025 scenarios and development path of intelligent coal mine,” Journal of China Coal Society, vol. 43, no. 2, pp. 295-305, 2018.

[2] Q. Lin, H. Zhang, X. Zhao, and M. Xu, “An overview of the development of open-pit driverless dump trucks,” Coal Engineering, vol. 53, no. 2, pp. 29-34, 2021.

[3] D. M. Bajany, X. Xia, and L. Zhang, “A MILP model for truck-shovel scheduling to minimize fuel consumption,” Energy Procedia, vol. 105, pp. 2739-2745, 2017.

[4] J. B. Mendes, M. F. S. V. D’Angelo, N. A. Maia, and R. R. Veloso, “A hybrid multiobjective evolutionary algorithm for truck dispatching in open-pit-mining,” IEEE Latin America Transactions, vol. 14, no. 3, pp. 1329-1334, March 2016.

[5] S. Zhang, C. Lu, S. Jiang, L. Shan, and N. N. Xiong, “An unmanned intelligent transportation scheduling system for open-pit mine vehicles based on 5G and big data,” IEEE Access, vol. 8, pp. 135524-135539, 2020.

[6] E. Topal and S. Ramazan, “A new MIP model for mine equipment scheduling by minimizing maintenance cost,” European Journal of Operational Research, vol. 207, no. 2, pp. 1065-1071, 2010.

[7] V. N. Coelho, M. J. F. Souza, I. M. Coelho, F. G. Guimaraes, T. Lust, and R. C. Cruz, “Multi-objective approaches for the open-pit mining operational planning problem,” Electronic Notes in Discrete Mathematics, vol. 39, pp. 233-240, December 2012.

[8] M. J. F. Souza, I. M. Coelho, S. Ribas, H. G. Santos, and L. H. C. Merschmann, “A hybrid heuristic algorithm for the open-pit mining operational planning problem,” European Journal of Operational Research, vol. 207, no. 2, pp. 1041-1051, December 2010.

[9] S. Zeng, R. Jiao, C. Li, X. Li, and J. S. Alkasassbeh, “A general framework of dynamic constrained multiobjective evolutionary algorithms for constrained optimization,” IEEE Transactions on Cybernetics, vol. 47, no. 9, pp. 2678-2688, 2017.

[10] K. Deb and H. Jain, “An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, Part I: solving problems with box constraints,” IEEE Transactions on Evolutionary Computation, vol. 18, no. 4, pp. 577-601, 2014.

[11] R. S. Sutton and A. G. Barto, “Reinforcement Learning: An Introduction,” Massachusetts: The MIT Press, 2018.

[12] B. Chen, C. Li, S. Zeng, S. Yang, and M. Mavrovouniotis, “An Adaptive Evolutionary Algorithm for Bi-Level Multi-objective VRPs with Real-Time Traffic Conditions,” 2021 IEEE Symposium Series on Computational Intelligence (SSCI), Orlando, FL, USA, pp. 1-8, 2021.