Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Burden Control Strategy Based on Reinforcement Learning for Gas Utilization Rate in Blast Furnace ⋆

Xiaoling Shen [*,**] **Jianqi An** [*,**,1] **Min Wu** [*,**]
Jinhua She [*,**,***]

[*] *School of Automation, China University of Geosciences, Wuhan 430074, China (e-mail: anjianqi@cug.edu.cn)*
[**] *Hubei Key Laboratory of Advanced Control and Intelligent Automation for Complex Systems, Wuhan 430074, China (e-mail: wumin@cug.edu.cn)*
[***] *School of Engineering, Tokyo University of Technology, Hachioji, Tokyo 192-0982, Japan(e-mail: she@stf.teu.ac.jp)*

Abstract: Gas utilization rate (GUR) is an important state parameter to reflect the energy consumption, the quality and production of the pig iron, and the distribution of the gas flow in a blast furnace. The GUR is mainly adjusted by burden distribution and hot-blast supply. According to the analysis of mechanism and data, burden distribution and hot-blast supply affect the GUR on a long-time scale and short-time scale, respectively. However, few of the previous researches proposed the control method for the GUR and they did not consider multi-time-scale characteristics. Thus, it is necessary to design a control strategy or system for the GUR considering the multi-time-scale characteristics, which can make the GUR have a reasonable development trend. This paper presented a burden control strategy based on a reinforcement learning algorithm for the GUR. The method improved the development trend of the GUR on a long-time scale. The experimental results demonstrated that the sequence of the parameters of the burden distribution given by the presented method ensured a reasonable development trend of the GUR on a long-time scale.

Keywords: Blast furnace, gas utilization rate, burden control strategy, reinforcement learning algorithm, long-time scale.

1. INTRODUCTION

A blast furnace (BF) is a complex reactor to convert iron ore into liquid pig iron through a series of physical changes and chemical reactions (shown in Fig. 1) (An et al., 2020, 2018b; Gomes et al., 2017). The iron ore and coke are discharged into a BF from the top to form the iron-ore layers and coke layers, which are controlled by burden distribution. The hot blast is discharged into a BF from the bottom, which is adjusted by hot-blast supply. The coke burns with the hot blast to form an upward gas flow. Then, the iron ore reacts with carbon monoxide in the upward gas flow to form liquid pig iron, slag, and BF gas flow. The BF gas flow is discharged from the top of a BF, which is called the top gas flow. The gas utilization rate (GUR) is the ratio of the carbon dioxide content to the total content of carbon monoxide and carbon dioxide in the top gas flow. The GUR, ρ_{CO}, is calculated as

$$\rho_{CO} = \frac{V_{CO_2}}{V_{CO} + V_{CO_2}}. \tag{1}$$

Improving the GUR is good for reducing consumption, improving the quality of the pig iron, and increasing the production. Thus, it is important to control the GUR.

Some researches analyzed the GUR based on mechanism analysis. For example, a definition of the GUR was given by analyzing the correlation between the GUR and the chemical reactions (Kou et al., 2016). A gas flow distribution and operation state were determined by the GUR in a BF (Xiang et al., 2013). An impact of the natural gas injection on GUR was analyzed by mathematical modeling and energy exchange (Guo et al., 2013).

In addition, some researches focused on predicting and/or optimizing GUR based on data-driven methods. For example, a relation model based on low-frequency feature extraction was established to analyze the explicit relation between the burden distribution and states (Zhang et al., 2017). Thereby, a decision-making strategy was designed to improve the GUR according to adjust the burden distribution (Wu et al., 2018). Besides, a hybrid model was established to improve the GUR according to analyzing the position of the pile surface (Shi et al., 2016). Meanwhile, some models were built to predict the GUR, such as a model based on an online sequential extreme learning machine (Li et al., 2017) and an echo state net-

⋆ This work was supported in part by the National Natural Science Foundation of China under Grants 61973287 and 61333002, and the 111 project under Grant B17040.
[1] Corresponding author: anjianqi@cug.edu.cn (Jianqi An)

Copyright lies with the authors

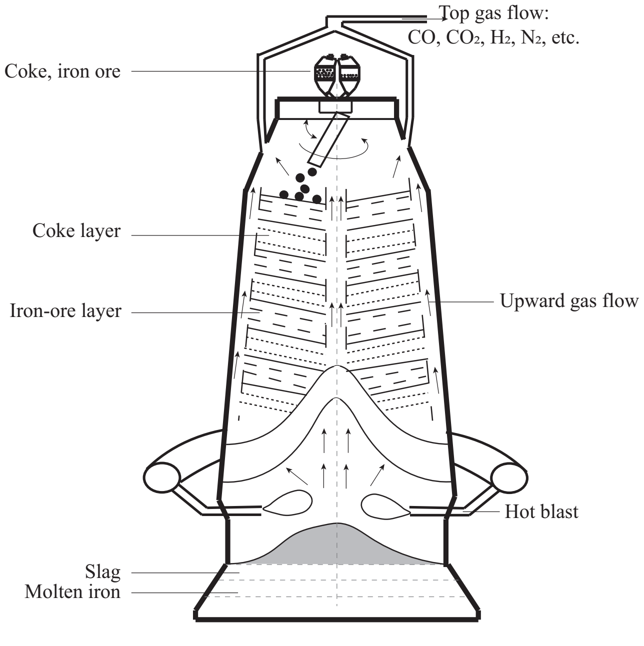

Fig. 1. Structure of BF

work ensemble model based on quantile regression (Lv et al., 2016). And some methods were proposed to analyze the characteristics of the GUR. For example, a method based on fuzzy theory was used to analyze the relationship between the top pressure and the gas distribution (An et al., 2018a). A model combining the T-S fuzzy neural network and particle swarm optimization was built to describe the relationship between operations and the GUR (Zhang et al., 2018). A model based on a fuzzy C-means clustering and statistical knowledge was used to analyze the features of the distribution for the gas flow center and corresponding GUR (Li et al., 2016). A phase space model based on the chaotic characteristics was built to reveal the characteristics of the development for the GUR (Xiao et al., 2017).

The previous researches made great contributions to optimizing the GUR, which provided some guidance for the BF operators. However, few of them gave the control method to improving the GUR. Besides, these methods only focused on a single-time scale when analyzing the relationship between the GUR and the operations of a BF. According to the mechanism and data analysis, burden distribution affects the GUR on a long-time scale; hot-blast supply affects the GUR on a short-time scale (An et al., 2019). Therefore, it is necessary to design a control method based on the multi-time-scale for the GUR.

The main purpose of this paper is to design a burden control strategy based on the reinforcement learning algorithm. The strategy is used to ensure that the GUR maintains a reasonable development trend on a long-term scale. While the high GUR means lower energy consumption, higher molten iron quality, and more reasonable gas flow distribution, which further means reasonable operations of a BF. Thus, the reasonable development trend in this paper is that the GUR keeps rising in a period of time. The rest of this paper is organized as follows. Section 2 introduces the reinforcement learning, especially the Q-learning algorithm. Section 3 presents a burden control strategy based on Q-learning on a long-time scale. Section 4 analyzes experimental results to demonstrate that the presented method improves the GUR on a long-time scale. And Section 5 draws conclusions and introduces the future works.

2. REINFORCEMENT LEARNING

Reinforcement learning learns a suitable behavior through the experiences generated by the interaction between an agent and the environment (Sutton and Barto, 2011). The formalization of reinforcement learning is based on a Markov decision process, which mainly contains 5 elements, $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}^{a(t)}_{s(t)s(t+1)}, \mathcal{R}(t+1), \gamma \rangle$, where

- $\mathcal{S}$ denotes a set of states of the environment, and $s(t) \in \mathcal{S}$ means a state at time t;
- $\mathcal{A}$ denotes a set of actions of an agent, and $a(t) \in \mathcal{A}$ means a action selected at time t;
- $\mathcal{P}^{a(t)}_{s(t)s(t+1)}$ denotes the probability of selecting $a(t)$ when $s(t)$ is changed to $s(t+1)$;
- $\mathcal{R}(t+1)$ denotes a reward given by the environment when $s(t)$ is changed to $s(t+1)$ by executing $a(t)$; and
- γ denotes the discount rate.

As the Q-learning algorithm is one of the most important algorithm in reinforcement learning proposed in Watkins and Dayan (1992), this paper uses a Q-learning algorithm to train the burden control sequence. The most important step of Q-learning is to train an action-value function $Q(s(t), a(t))$, which means the expected value when $s(t)$ is changed to $s(t+1)$ after executing $a(t)$. $Q(s(t), a(t))$ is updated as

$$\begin{cases} Q(s(t), a(t)) = Q(s(t), a(t)) + \alpha \Delta \mathcal{R} \\ \Delta \mathcal{R} = \mathcal{R}(t+1) + \gamma \max_{a} Q(s(t+1), a) - Q(s(t), a(t)) \end{cases}, \tag{2}$$

where $\alpha \in (0, 1]$ is the learning rate.

According to the updated method of $Q(s(t), a(t))$, the Q-learning considers the impact of current state and action on subsequent states. In the iron-making process, the subsequent GUR is affected by the current parameters of the operations of BF. Thus, this paper uses the Q-learning algorithm to design the burden control strategy for the GUR on a long-time scale.

3. BURDEN CONTROL STRATEGY FOR GUR

This section introduces a burden control strategy for the GUR based on Q-learning algorithm, which is designed to keep a reasonable development trend for the GUR on a long-time scale. The method contains 6 parts: the selection of the controlled state $G(t)$, the definition of states $\mathcal{S}$ and actions $\mathcal{A}$, the policy of action selection $\pi(a(t)|s(t))$, the calculation of the reward $\mathcal{R}$, the update of $Q(s(t), a(t))$, and the strategy of burden control.

3.1 Selection of Controlled State

The main purpose of the method is to keep the GUR rising on the long-time scale. Thus, this paper uses the long-time-scale part of the GUR as the controlled state

$G(t)$. The long-time-scale part of the GUR is obtained by the decomposition and reconstruction methods proposed in An et al. (2019).

3.2 Definition of States and Actions

This part defines the states and actions used in the burden control strategy.

States The goal of the control strategy is to improve the development of the long-time-scale part of the GUR. According to the analysis in An et al. (2019), the long-time-scale part of the GUR is mainly affected by the burden distribution. Thus, the parameters of the burden distribution are selected as the states in $\mathcal{S}$.

According to the mechanism relationship, the changes in the parameters of the burden distribution are directly reflected in the ore-to-coke ratios and central-coke ratios in a BF. The ore-to-coke ratio is a ratio of the thickness of the ore layer to the coke layer in the corresponding angular position; the central-coke ratio, the ratio of the amounts of the central coke to the total coke in the corresponding angular position. Thus, this paper uses 4 ore-to-coke ratios and 2 central-coke ratios as the states of burden distribution according to the actual run data collected from a 2800 m^3 BF.

Besides, the long-time part of the GUR cannot keep rising for an indefinite period of time. Therefore, this paper adds a state of time to $\mathcal{S}$, which means it keeps rising for a certain period of time.

Hence, the state, $s(t) \in \mathcal{S}$, is defined as

$$\begin{aligned} s(t) = (&s_T(t), s_{oc1}(t), s_{oc2}(t), s_{oc3}(t),\\ &s_{oc4}(t), s_{c1}(t), s_{c2}(t)), \end{aligned} \tag{3}$$

where s_T is used to control the total period of time.

Since the real values of the ore-to-coke ratio and central-coke ratio vary at all times, this paper uses the interval to represent the states of burden distribution. For each ore-to-coke ratio and central-coke ratio, this paper first divides the real values into different intervals by a frequency distribution histogram. Then, this paper chooses the median value of the interval to represent this interval. $\mathcal{S}$ contains these median values. The states in $\mathcal{S}$ are represented by these median values.

The interval of each state is different. In order to calculate the next state $s(t+1)$, this paper first numbers each interval from small to large for each state of burden distribution (n_*). Then, the state $s(t)$ is mapped to $\widehat{s}(t)$:

$$\begin{aligned} \widehat{s}(t) = (&s_T(t), n_{oc1}(t), n_{oc2}(t), n_{oc3}(t),\\ &n_{oc4}(t), n_{c1}(t), n_{c2}(t)), \end{aligned} \tag{4}$$

where $\widehat{n}_*(t)$ means the interval number corresponding to $s_*(t)$.

Actions According to the state, the action contains two parts. One part is used to adjust the states of time that the value of the operation is 1. And the other part is used to adjust the states of the burden distribution that each operation has 3 values: 1 (up), 0 (unchanged), -1(down). As the state contains 6 parameters of burden distribution, the action should contain 6 corresponding operations. Then the selection of action is 3^6.

In order to reduce the complexity of the algorithm, this section analyzes the correlations between each parameters of burden distribution based the Pearson coefficient method. The correlation is calculated as

$$\gamma_{jk} = \frac{\sum_{t=1}^{L}(x_k(t)-\bar{x}_k)(x_j(t)-\bar{x}_j)}{\sqrt{\sum_{t=1}^{L}(x_k(t)-\bar{x}_k)^2(x_j(t)-\bar{x}_j)^2}}, \tag{5}$$
$$k = 1, 2, \cdots, 6;\ j = 1, 2, \cdots, 6,$$

where γ_{jk} is the correlation between $x_k(t)$ and $x_j(t)$; L, the length of the time series of $x_k(t)$ and $x_j(t)$; $x_k(t)$ and $x_j(t)$ are the time series of the kth and jth states of burden distribution, respectively; $\bar{x}_k$ and $\bar{x}_j$, the average values of $x_k(t)$ and $x_j(t)$, respectively. The larger γ_{jk} is, the stronger correlation between $x_k(t)$ and $x_j(t)$ is.

Table 1 shows the correlation between each state of burden distribution. It is clear that s_{oc1}, s_{oc2}, and s_{oc3} are positively correlated, which means they can be adjusted by an operation. s_{c1} and s_{c2} are negatively correlated, which means they can also be adjusted by an operation, just by using the opposite value. s_{oc4} is not related to others.

Table 1. Correlation between states of burden distribution

States	s_{oc1}	s_{oc2}	s_{oc3}	s_{oc4}	s_{c1}	s_{c2}
s_{oc1}	1.000	**0.940**	**0.846**	-0.024	-0.187	0.300
s_{oc2}	**0.940**	1.000	**0.837**	-0.017	-0.205	0.316
s_{oc3}	**0.846**	**0.837**	1.000	0.041	-0.381	0.348
s_{oc4}	-0.024	-0.017	0.041	1.000	-0.204	0.231
s_{c1}	-0.187	-0.205	-0.381	-0.2044	1.000	**-0.746**
s_{c2}	0.300	0.317	0.348	0.231	**-0.746**	1.000

Based on the correlation, the action, $a(t) \in \mathcal{A}$, is defined as

$$a(t) = (a_T(t), a_1(t), a_2(t), a_3(t)). \tag{6}$$

Thus, the next state $s(t+1)$ after executing $a(t)$ in $s(t)$ is calculated as follows:

$$\begin{aligned} \widehat{s}(t+1) = (&s_T(t+1), n_{oc1}(t+1), n_{oc2}(t+1), n_{oc3}(t+1),\\ &n_{oc4}(t+1), n_{c1}(t+1), n_{c2}(t+1)), \end{aligned} \tag{7}$$

where

$$\begin{cases} s_T(t+1) = s_T(t) + a_T(t) \\ n_{oc1}(t+1) = n_{oc1}(t) + a_1(t) \\ n_{oc2}(t+1) = n_{oc2}(t) + a_1(t) \\ n_{oc3}(t+1) = n_{oc3}(t) + a_1(t) \\ n_{oc4}(t+1) = n_{oc4}(t) + a_2(t) \\ n_{oc5}(t+1) = n_{oc5}(t) + a_3(t) \\ n_{oc6}(t+1) = n_{oc6}(t) - a_3(t) \end{cases}. \tag{8}$$

Then,

$$\begin{aligned} s(t+1) = (&s_T(t+1), s_{oc1}(t+1), s_{oc2}(t+1), s_{oc3}(t+1),\\ &s_{oc4}(t+1), s_{c1}(t+1), s_{c2}(t+1)), \end{aligned} \tag{9}$$

where $s_*(t+1)$ is the median value of the interval corresponding to $n_*(t+1)$.

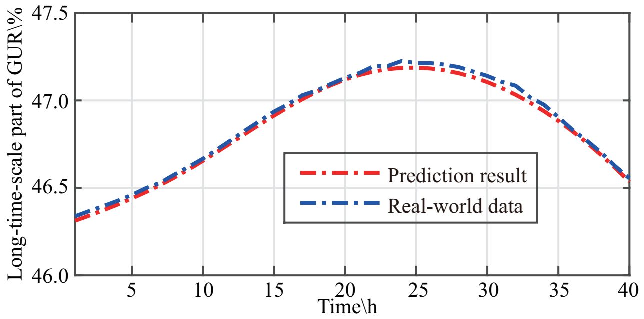

Fig. 2. Prediction of GUR on long-time scale

3.3 Policy of Action Selection

This paper takes a ϵ-greedy algorithm as the policy of action selection, which is shown as

$$\pi(a(t) \mid s(t)) = \begin{cases} \epsilon/m + 1 - \epsilon & \widetilde{a} = \arg\max_{a \in \mathcal{A}} Q(s(t), a) \\ \epsilon/m & \text{else} \end{cases}, \tag{10}$$

where ϵ is exploratory rate; $\widetilde{a}$, the action that maximize $Q(s(t), a)$; m, the number of actions.

Thus, this paper uses the probability of $\epsilon/m+1-\epsilon$ to select the action that maximize $Q(s(t), a)$ and the probability of ϵ/m to randomly select the other actions.

3.4 Calculation of Reward

In order to ensure a reasonable development trend of the GUR on a long-time scale, the reward $\mathcal{R}(t+1)$ is designed as

$$\mathcal{R}(t+1) = \widehat{G}(t+1) - \widehat{G}(t), \tag{11}$$

where $\widehat{G}(t+1)$ and $\widehat{G}(t)$ are the prediction value of the long-time part of the GUR at time $t+1$ and t, respectively, which are obtained by the long-time-scale prediction model of the GUR.

This paper uses the BP neural network algorithm to establish the long-time-scale prediction model. The parameters when training the model are as follows: hidden layers is 9; epochs, 3000; goal, $10^{(-3)}$; lr, 0.1; transfer function, *tansig*; training function, *trainlm*; bias learning function, *learngdm*. The inputs of the prediction model contain two parts: 6 states of burden distribution and 6 historical information of the GUR. The number of history information is calculated by the partial autocorrelation function in Box and Jenkins (1971). The output of the prediction model is the long-time-scale part of the GUR. The long-time-scale part of the GUR is calculated by the methods shown in An et al. (2019).

This paper uses four-month continuous samples of real-world industrial data that were selected from the database of a 2800 m^3 BF. 450 samples are used for training; the rest 40 samples, for testing. The interval time between samples is 6 hours. Figure 2 shows that the prediction model accurately predicts the development trend of GUR on a long-time scale, which can be used to calculate the reward.

3.5 Update of Action-Value Function

The aim of the update is to train a table of $Q(s(t), a(t))$ by multiple iterations. For each iteration, $Q(s(t), a(t))$ is calculated based on $\pi(a(t) \mid s(t))$ and $\mathcal{R}(t+1)$ by (2). The update will not stop until the maximum number of iterations or $Q(s(t), a(t))$ converges.

3.6 Strategy of Burden Control

The presented method designs a strategy of burden control according to the table of $Q(s(t), a(t))$. The strategy is obtained as follows:

Step 1 Select the action $a(t)$ as

$$a(t) = \arg\max_{a \in \mathcal{A}} Q(s(t), a). \tag{12}$$

Step 2 Calculate $\widehat{s}(t+1)$ based on $s(t)$ and $a(t)$ by (7) and (8).

Step 3 Calculated the state sequence of burden distribution in $s(t+1)$ by using the interval number obtained by $\widehat{s}(t+1)$.

Step 4 Get the burden control strategy based on the state sequence of burden distribution.

4. EXPERIMENT AND DISCUSS

This section proves the effectiveness of the presented method by comparing the experimental result and the real-world industrial data. The data used in this experiment were selected from the database of a 2800 m^3 BF.

In the experiment, the maximum value of s_T is 12. As the interval time between samples is 6 hours, the total period of time is 72 hours (3 days). Besides, the current prediction value of the long-time-scale part of the GUR is taken as history information in the next time, which is used as one of the inputs in the long-time-scale prediction model.

Figure 3 shows the state sequences of burden distribution, which is the burden control strategy. Each data is a median value of an interval, which represented an interval. Table 2 shows the intervals corresponding to the data shown in Fig. 3. The data in time 1 is the initial value of each parameter. From the results, s_{oc1}, s_{oc2}, and s_{oc3} are basically the same as the initial values on the whole. s_{oc4} is higher than its initial value. s_{c1} is lower than its initial value and s_{c2} is higher than its initial value.

Figure 4 shows the comparison of the experimental results and the real-world data of the GUR on a long-time scale. The experiment results (the red line) are obtained by the sequences of the parameters of the burden distribution (shown in Fig. 3) and the prediction model of the GUR on the long-time scale. The results are higher than real-world data. Besides, the results show that the burden control strategy changes the downward trend of the GUR on the long-time scale. Figure 5 shows that the burden control strategy increases the long-time-scale part of the GUR by 1.5% at most. The experiment demonstrates the effectiveness of the presented method.

Figure 6 shows the comparison of the fusion result and the real-world data of the GUR. The fusion result is calculated by the long-time-scale part and the short-time-scale part of the GUR. The long-time part of the GUR is calculated by

Table 2. Intervals corresponding to burden control strategy

Time	s_{oc1}	s_{oc2}	s_{oc3}	s_{oc4}	s_{c1}	s_{c2}
1	(7.104,7.328]	(6.918,7.145]	(7.195,7.450]	(4.451,4.859]	(0.072,0.075]	(0.263,0.276]
2	(7.104,7.328]	(6.918,7.145]	(7.195,7.450]	(4.451,4.859]	(0.070,0.072]	(0.276,0.290]
3	(7.104,7.328]	(6.918,7.145]	(7.195,7.450]	(4.042,4.451]	(0.067,0.070]	(0.290,0.304]
4	(7.104,7.328]	(6.918,7.145]	(7.195,7.450]	(4.042,4.451]	(0.065,0.067]	(0.304,0.318]
5	(7.104,7.328]	(6.918,7.145]	(7.195,7.450]	(4.042,4.451]	(0.062,0.065]	(0.318,0.331]
6	(7.104,7.328]	(6.918,7.145]	(7.195,7.450]	(4.451,4.859]	(0.059,0.062]	(0.331,0.345]
7	(7.104,7.328]	(6.918,7.145]	(7.195,7.450]	(4.859,5.268]	(0.059,0.062]	(0.331,0.345]
8	(6.879,7.104]	(6.691,6.918]	(6.940,7.195]	(4.451,4.859]	(0.059,0.062]	(0.331,0.345]
9	(7.104,7.328]	(6.918,7.145]	(7.195,7.450]	(4.859,5.268]	(0.059,0.062]	(0.331,0.345]
10	(6.879,7.104]	(6.691,6.918]	(6.940,7.195]	(5.268,5.676]	(0.059,0.062]	(0.331,0.345]
11	(6.879,7.104]	(6.691,6.918]	(6.940,7.195]	(4.859,5.268]	(0.062,0.065]	(0.318,0.331]
12	(7.104,7.328]	(6.918,7.145]	(7.195,7.450]	(4.859,5.268]	(0.062,0.065]	(0.318,0.331]

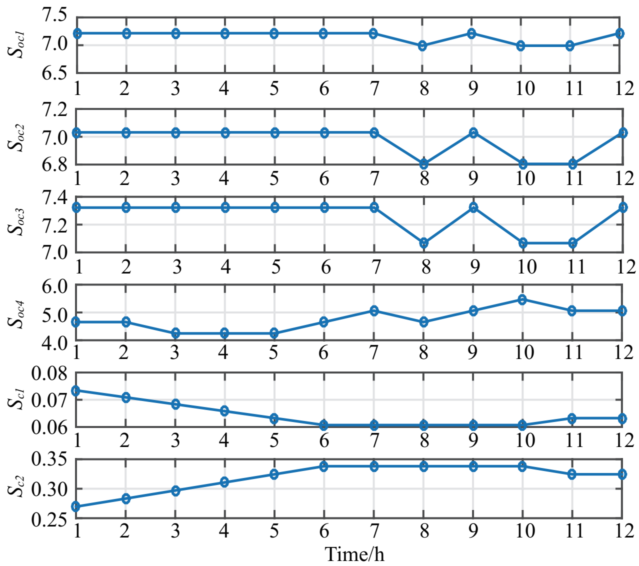

Fig. 3. State sequences of burden distribution

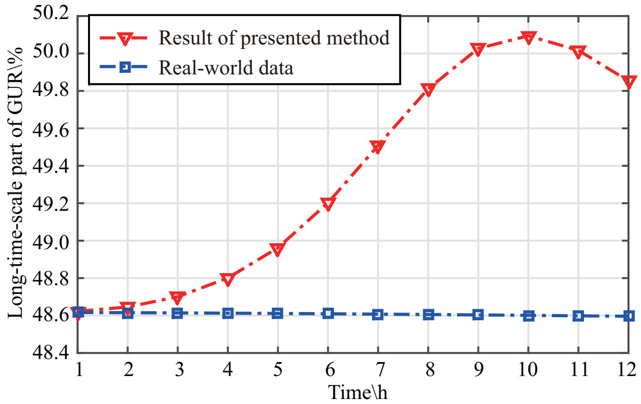

Fig. 4. Comparison of experimental result and real-world data of GUR on long-time scale

the method presented in this paper. The short-time part is the real-world data of the GUR controlled by hot-blast supply. The result shows that the burden control strategy improves the value of the GUR as a whole by increasing the long-time-scale part of the GUR, further illustrating the effectiveness of the presented method.

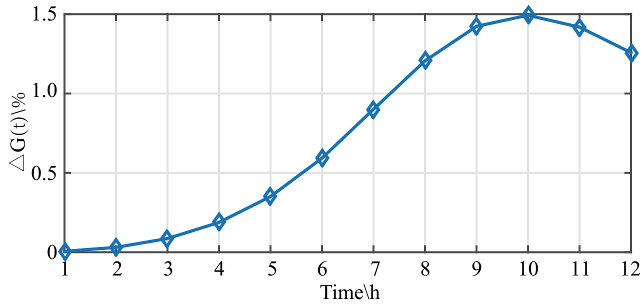

Fig. 5. Distance between result of presented method and real-world data

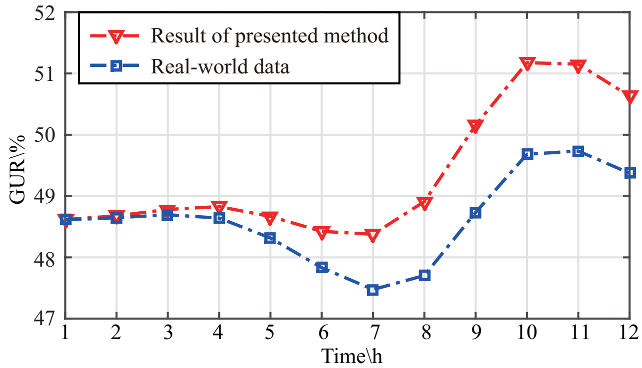

Fig. 6. Comparison of fusion result and real-world data of GUR

5. CONCLUSION

The main contribution of this paper is to present a burden control strategy based on Q-learning algorithm for the GUR. The goal of the method is to make a reasonable development trend of the GUR on a long-time scale. The experiment demonstrates that the presented method yields a state sequence of burden distribution, which increases the GUR on the long-time scale.

The method presented in this paper improves the GUR and changes its development trend. Meanwhile, this method uses the correlations between each state to design the actions, which reduces the complexity of the method. However, this method only considers a fixed step size when training the burden control strategy. Besides, the method only gives the sequences of the ore-to-coke ratios and central-coke ratios that reflect the changes in the operating parameters of the burden distribution. Thus, we will use a random step size when finding the burden control strategy. And we will study the relationship between the ore-to-coke and central-coke ratios and the operation parameters.

REFERENCES

An, J., Huang, Y., Wu, M., and She, J. (2020). Two-level data-based adjustment of controller parameters for weighing process of ladle furnace. *IEEE Transactions on Industrial Informatics*, DOI: 10.1109/TII.2020.2989160.

An, J., Shen, X., Wu, M., and She, J. (2019). A multi-time-scale fusion prediction model for the gas utilization rate in a blast furnace. *Control Engineering Practice*, 92, DOI: 10.1016/j.conengprac.2019.104120.

An, J., Yang, J., Wu, M., She, J., and Terano, T. (2018a). Decoupling control method with fuzzy theory for top pressure of blast furnace. *IEEE Transactions on Control Systems Technology*, 27(6), 2735–2742.

An, J., Zhang, J., Wu, M., Cao, W., and Terano, T. (2018b). Soft-sensing method for slag-crust state of blast furnace based on two-dimensional decision fusion. *Neurocomputing*, 315, 405–411.

Box, G.E.P. and Jenkins, G.M. (1971). Time series analysis, forecasting and control. *Journal of the American Statistical Association*, 134(3).

Gomes, F.S.V., Coco, K.F.J., and Salles, L.F. (2017). Multistep forecasting models of the liquid level in a blast furnace hearth. *IEEE Transactions on Automation Science & Engineering*, 14(2), 1286–1296.

Guo, T.L., Chu, M.S., Liu, Z.G., Tang, J., and Yagi, J.I. (2013). Mathematical modeling and exergy analysis of blast furnace operation with natural gas injection. *Steel Research International*, 84(4), 333–343.

Kou, M., Wang, L., Xu, J., Wu, S., and Cai, Q. (2016). Low co2 emission by improving co utilization ratio in chinas blast furnaces. *Ironmaking and Steelmaking Processes*, 199–212.

Li, S., Wen, Y.B., Zhao, G.S., and Yu, T. (2016). Recognition of blast furnace gas flow center distribution based on infrared image processing. *Journal of Iron and Steel Research, International*, 23(3), 203–209.

Li, Y., Zhang, S., Yin, Y., Xiao, W., and Zhang, J. (2017). A novel online sequential extreme learning machine for gas utilization ratio prediction in blast furnaces. *Sensors*, 17(8), 1847–1870.

Lv, Z., Zhao, J., Liu, Y., and Wang, W. (2016). Use of a quantile regression based echo state network ensemble for construction of prediction intervals of gas flow in a blast furnace. *Control Engineering Practice*, 46, 94–104.

Shi, L., Zhao, G., Li, M., and Ma, X. (2016). A model for burden distribution and gas flow distribution of bell-less top blast furnace with parallel hoppers. *Applied Mathematical Modelling*, 40(23-24), 10254–10273.

Sutton, R. and Barto, A.G. (2011). Reinforcement learning: An introduction. *MIT Press*.

Watkins, C.J.C.H. and Dayan, P. (1992). Technical note: Q-learning. *Machine Learning*, (3-4), 279–292.

Wu, M., Zhang, K., An, J., She, J., and Liu, K. (2018). An energy efficient decision-making strategy of burden distribution for blast furnace. *Control Engineering Practice*, 78, 186–195.

Xiang, Z.Y., Wang, X.L., and Han, Y. (2013). More discussion on evaluation method for productive efficiency of ironmaking blast furnace. *Iron & Steel*, 48, 86–91.

Xiao, D., An, J., He, Y., and Wu, M. (2017). The chaotic characteristic of the carbon-monoxide utilization ratio in the blast furnace. *ISA Transactions*, 68, 109–115.

Zhang, K., Wu, M., An, J., Cao, W., Liu, Z., and Ning, F. (2017). Relation model of burden operation and state variables of blast furnace based on low frequency feature extraction. *IFAC PaperOnLine*, 50(1), 13796–13801.

Zhang, S., Jiang, H., Yin, Y., Xiao, W., and Zhao, B. (2018). The prediction of the gas utilization ratio based on ts fuzzy neural network and particle swarm optimization. *Sensors*, 18(2), 625–645.