

Article

Deep Q-Network for Optimal Decision for Top-Coal Caving

Yi Yang ¹ , Xinwei Li ^{1,*}, Huamin Li ², Dongyin Li ² and Ruifu Yuan ²

¹ School of Electrical Engineering and Automation, Henan Polytechnic University, Jiaozuo 454000, China; yangyi@hpu.edu.cn

² School of Energy Science and Engineering, Henan Polytechnic University, Jiaozuo 454000, China; lihuamin2007@163.com (H.L.); lidongyin@126.com (D.L.); yrf@hpu.edu.cn (R.Y.)

* Correspondence: lixinwei@hpu.edu.cn; Tel.: +86-391-3978551

Received: 21 January 2020; Accepted: 24 March 2020; Published: 2 April 2020



Abstract: In top-coal caving, the window control of hydraulic support is a key issue to the perfect economic benefit. The window is driven by the electro-hydraulic control system whose command is produced by the control model and the corresponding algorithm. However, the model of the window's control is hard to establish, and the optimal policy of window action is impossible to calculate. This paper studies the issue theoretically and, based on the 3D simulation platform, proposes a deep reinforcement learning method to regulate the window action for top-coal caving. Then, the window control of top-coal caving is considered as the Markov decision process, for which the deep Q-network method of reinforcement learning is employed to regulate the window's action effectively. In the deep Q-network, the reward of each step is set as the control criterion of the window action, and a four-layer fully connected neural network is used to approximate the optimal Q-value to get the optimal action of the window. The 3D simulation experiments validated the effectiveness of the proposed method that the reward of top-coal caving could increase to get a better economic benefit.

Keywords: top-coal caving; deep reinforcement learning; deep Q-network; discrete element method; 3D-simulation

1. Introduction

Coal is one of the most important energy sources in the world [1]. Even though its consumption has decreased in the past years, coal will persist in the domination of primary energy for the next several decades [2–4]. Improving the coal mining technology sustainably to alleviate the damage to the environment is the preferred choice for countries that lack oil sources while rich in coal [5,6]. Underground mining the thick coal seam is the most economical and environmentally-friendly mode currently [7,8]. In the longwall workspace, the top-coal caving is the most effective technology to exploit the coal seam whose thickness is greater than 4 m [9–11]. Especially in China, more than 40% coal is in the thick coal seam; hence, top-coal caving is the development direction in the future years [12].

By this method, the coal seam is first cut by coal cutter at, then the rest of the top-coal falls by the combined action of gravity and pressure, and the falling coal is captured by pulling back the tail canopy of the hydraulic support. Therefore, the hundreds of hydraulic supports in the workspace are the key devices for the security and top-coal capturing, and controlling the tail canopy of hydraulic supports optimally is the critical issue for top-coal caving [13,14].

The tail-beam, named “window” for capturing the top-coal, opens to capture the falling coal and closes to avoid the falling rocks as much as possible [15]. The traditional control method of the

window is manual manipulation based on the experience, and the most important criterion is to close the window if the rock emerges.

In recent years, top-coal caving automatically and intelligently is an important research topic [16] in which the optimal control of windows' action is the key issue for the regulation of tail-beam. That means there should be an optimal control algorithm to regulate the electro-hydraulic control system that drives the hydraulic supports. In classical control theory, the optimal control algorithm is designed based on the model of the controlled plant. Hence, the model of the top-coal caving process should be established first. Unfortunately, the process is so complex that the dynamic equations about the top-coal caving are hard or even impossible to obtain. This is one of the key issues for automating the windows actions.

For this reason, at present, the research of top-coal caving focuses on optimizing the process technology and most of it used 2D simulation based on *discrete element method* (DEM) [17]. In [18,19], the particle flow code is used to research the effect of top-coal thickness on top-coal recovery ratio under the caving mining technique. Wang et al. [20] employed the DEM to validate the boundary-body-ratio of top-coal caving. In [21], the relationship between mining heights and shield resistance in the longwall panel is researched by DME simulation. In [22], EDM is used to analyze the top-coal recovery.

Furthermore, to obtain a better performance, several different kinds of technological processes of top-coal caving are proposed to improve the coal recovery. The typical works are introduced in [15,17,23,24]. The core idea of the method is to control the boundary between the rock layer and the coal layer by operating the windows' action in a multi-round. During the top-coal caving, the boundary should be as straight as possible so that the coal and rock can be separated easily by setting the open time of the windows [23].

The aforementioned references show that at present the research about the optimal strategy of top-coal caving focuses on the technological process. However, as a control system, the input of the windows operation decision is just the open time of the window, and the decision mechanism is "close the window if the rock emerges". It is well known that the top-coal movement depends on a complex dynamic process [25,26] in which the geologic structure [14,27], roof pressure [10], hydraulic support state [28], etc. are coefficients to the performance of top-coal caving [29–31]. That means the huge coal may be on the top of some rock, especially if there is a rock layer in the coal seam. Hence, if we hope to capture the top-coal accurately, more information should be introduced into the decision system, and an optimal decision mechanism should be designed based on that information.

The window executing action over time is a typical Markov process [32]. The action decision is regulated by a given mechanism based on the environment state, while the Markov property of the action decision processes holds that the current action only depends on the last process. *Reinforcement learning* (RL) [33] is an effective methodology to deal with the optimal decision of the Markov process. It learns the regulation mechanism by the reward of the agent iteratively. Especially, the model-free method of RL named Q-learning [34,35] can learn the action-value function without needing a control model; then, the optimal action can be calculated based on the action-value function. Hence, the Q-learning method can avoid establishing the control model in the classical control theory.

Nevertheless, in top-coal caving, the state of the environment is a continuous parameter, such as the thickness of the coal seam, the pressure of the hydraulic support, and the ratio of the rock in the coal. Hence, the action-value function of the Q-learning is difficult to be depicted by a simple state-action table and hard to learn the value.

The rapid rise of artificial intelligence [36], especially the deep neural network [37–39], provides another powerful instrument to approximate the nonlinear system with a high dimension. As a matter of course, the deep neural network is employed to approximate the action-value function of Q-learning. It is a new branch of interdisciplinary research called deep reinforcement learning [32,40–43] and obtains great successes in autonomous driving [44], robot control [45], multi-task, multi-agent [46,47], etc.

In this paper, a new 3D DEM simulation test platform based on the open-source framework YADE [48–50] is developed to analyze the dynamic process of top-coal caving. To get the optimal decision of windows intelligently based on the simulation platform, this paper, along with our preliminary work [51], introduces more information of windows action during top-coal caving as the state of the control system and employs the deep Q-network method of reinforcement learning to approximate the windows optimal decision. The main contributions of this work include:

- (1) The optimal control of the window's action of hydraulic support is transformed into a Markov decision process and a new method based on deep Q-network is proposed to regulate the optimal decision of the window's action. In the method, the state of the environment, the loss function of the optimizer, and the reward of each step are given according to the process of top-coal caving.
- (2) A 3D discrete element method simulation platform is created to analyze the process of top-coal caving based on Yade. Based on the simulation platform, simulation experiments were carried out and the results theoretically validate an available way of applying the intelligent method to top-coal caving.

The rest of the article is organized as follows. In Section 2, the 3D simulation platform is introduced. The optimal decision of top-coal caving by deep reinforcement learning is presented in Section 3. In Section 4, the experiment and result analyses are given. The conclusion is shown in Section 5.

2. Top-Coal Caving 3D Simulation Platform

Most top-coal caving simulations about the optimal decision of windows action is based on DEM in two dimensions, as shown in Figure 1. For this simulation, the boundary between rock and coal can be shown clearly, and the effect of the drawing coal could be analyzed directly.

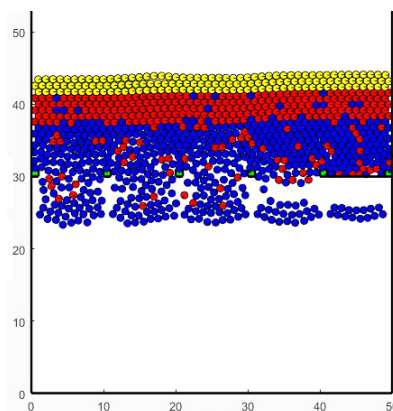


Figure 1. Top-coal caving 2D simulation based on DEM. This is our early work developed by Matlab [51]. The particles in blue, red, and yellow are coal, immediate roof, and basic roof, respectively. In this simulation platform, the controller gets the states of the system, and then calculates the optimal decision of windows' action by the design algorithm. The source code can be accessed at: "<https://github.com/YangYi-HPU/Reinforcement-learning-simulation-environment-for-top-coal-caving>" (for non-commercial uses only).

In most 2D simulations, the window executing the action often ignores the process, i.e., the window changes the state from close to open instantaneously. Hence, even though the regulation mechanism of the windows' action is only based on "close the window if the rock emerges", it can get a good result. However, in practice, it takes time to open and close the window. If the rock emerges near the window, the closing process would make the rock fall into the drag conveyor. Furthermore, 2D simulation fails to reflect the real scenario, especially when the particles are lying on a plane that the movement of the coal on shield-beam cannot depict, and the real boundary between

coal and rock is not a line but a plane. Hence, the method to control the boundary should expand to adjust the flatness of the plane.

Yade is a well known open-source framework driven by python for DEM [49,52] in the Linux operation system. It is flexible to integrate the control algorithm of the window action with the complex calculation. In this paper, we focus on the windows optimal control from a control system perspective; hence, we only demonstrate five windows for the process of top-coal caving. The three scenarios are shown in Figure 2.

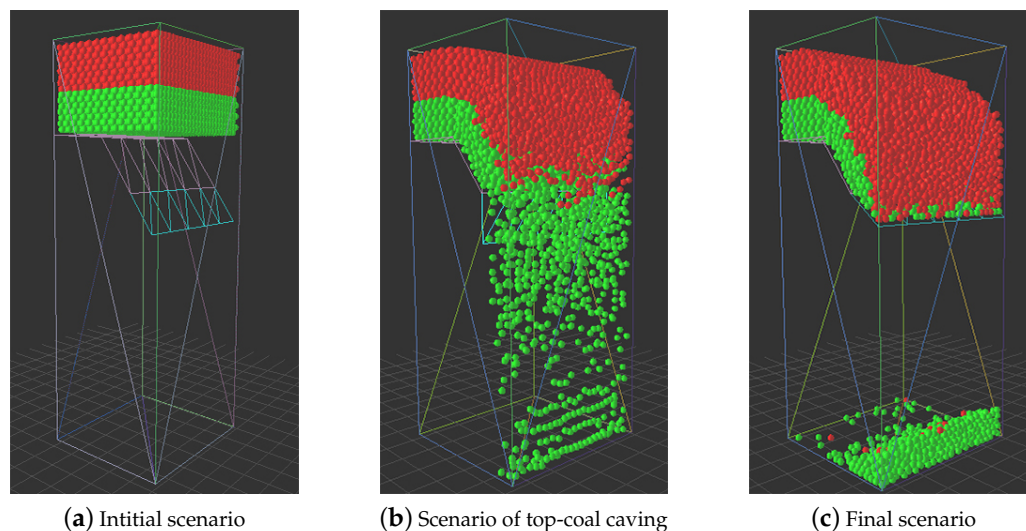


Figure 2. Scenarios of top-coal caving. The read and green particles are, respectively, the rock and coal. (a) In the initial scenario, the windows are closed. (b) In the second scenario, the particles collapse under the gravity. (c) In the final scenario, the windows close regulated by a given mechanism.

There are five hydraulic supports simplified to the top-beam, shield-beam, and tail-beam. The tail-beam can rotate around the bottom of shield-beam with the given speed, and the scope of swaying is restricted by lower and upper bounds. The parameters of the simulation platform are shown in Table 1.

Table 1. Parameters of 3D top-coal caving simulation platform.

w_{sp}	w_{hy}	h_{hy}	l_{sh}	l_{ta}	θ_s	θ_u	θ_l
6.8 m	1.5 m	3.8 m	3 m	2 m	50°	15°	45°

In Table 1, w_{sp} is the width of workspace, w_{hy} is the width of hydraulic support, h_{hy} is the height of hydraulic support, l_{sh} is the length of shield-beam, l_{ta} is the length of tail-beam, θ_s is the angle of between shield-beam and top beam, θ_u is the upper angle between tail-beam and shield-beam, and θ_l is the lower angle between tail-beam and shield-beam. The height of the rock layer and coal layer can be set as the requirement. The other parameters such as height of the space, the boundary of environment, and the location of each hydraulic support can be calculated according to the geometric relationship.

It should be noted that the parameters of the simulation are set as close to the real situation as possible; however, some parameters such as the height of rock could not set the same as the real case because the huge calculation of DEM may take several days. Meanwhile, this scheme does not hinder the validation of the algorithm.

In this platform, the material properties of rock and coal are shown in Table 2. They are set according to Tashan coal mine in China.

Table 2. Particle material properties of the simulation.

	Young Modulus (Pa)	Cohesion (Pa)	Density (kg/m ³)	Friction Angle (°)	Poisson Rate	Tensile Strength (Pa)	Normal Stiffness (Pa)	Shear Stiffness (Pa)
coal	2×10^8	2.06×10^6	1373	44.82	0.29	6.9×10^5	1.5×10^6	1.13×10^6
rock	4×10^8	2.11×10^6	2542	33.6	0.23	1.51×10^6	15.1×10^6	1.13×10^6

3. Optimal Decision of Top-Coal Caving with Deep Q-Network

3.1. Markov Process of Top-Coal Caving

The process of windows executing action is a time series and meets the Markov property. The window action decision is considered as a control system; the input is the environment state and the output is the window action. Hence, the optimal control of window action is a Markov decision process essentially.

Given the state space of the top-coal caving denoted by $\mathbb{S} = \{s_1, s_2, \dots, s_n\}$, s_i is the state of environment, $i = 1, 2, \dots, n$; n is the dimension of state space. For top-coal caving; and s_i is a continuous variable. The window action space is $\mathbb{A} = \{a_1, a_2, \dots, a_m\}$; a_j is the discrete action value, $j = 1, 2, \dots, m$; and m is the dimension of action. The Markov decision process is denoted by $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}, \gamma\}$, where $\mathcal{S} \in \mathbb{S}$, $\mathcal{A} \in \mathbb{A}$, and $\mathcal{R}$ is the reward of action under the given state. $\mathcal{P}$ is the transition probability distribution of the current state to next state with the action. γ is the discount factor, $\gamma \in (0, 1)$.

The policy π is the function of state s . There are two ways to get policy value. The first is deterministic policy, $a = \pi(s)$. It indicates that, if the state is s , the action a must be chosen as the deterministic value. The second is probabilistic policy, denoted by $\pi(a|s)$. That means the probability of executing action a under the state s .

According to dynamic programming [33,53], define the value function $v^\pi(s)$ at point t with the state s under the given policy π .

$$v^\pi(s) = \mathbb{E}_\pi \left(\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | s_t = s \right) \quad (1)$$

The purpose of reinforcement learning is to find an optimal policy that can get the maximum value of the $v^\pi(s)$ by Equation (1). More formally, the action-value function is defined based on $v^\pi(s)$, i.e., the cumulative rewards at point t if the action is chosen as a . Action-value is formalized as Equation (2).

$$Q^\pi(s, a) = \mathbb{E}_\pi \left(\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | s_t = s, a_t = a \right) \quad (2)$$

By Equations (1) and (2), the optimal policy can be calculated from the optimal action-value function shown as Equation (3).

$$Q^*(s, a) = \max_{\pi} \mathbb{E}_\pi \left(\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} | s_t = s, a_t = a \right) \quad (3)$$

The deterministic policy to get optimal action is shown as Equation (4) if the state is s .

$$a^* = \arg \max_a Q^*(s, a) \quad (4)$$

Hence, to get the optimal action of the window, the action-value function $Q(s, a)$ should be trained. If the state is a discrete variable, $Q(s, a)$ could be considered as the table that depicts the

relation between state space and action space. The Q-learning algorithm is a mechanism to train $Q(s, a)$. The $Q(s, a)$ iteration of Q-learning shown as (5).

$$Q(s, a) \leftarrow (1 - \alpha) Q(s, a) + \alpha \left(R(s, a) + \gamma \max_a Q(s', a) \right) \quad (5)$$

where s' is the next state and α is the learning rate.

However, for the continuous state, it is hard or even impossible to show all the states in a table due to the dimension curse [33]. Fortunately, the deep neural network [36,37,39] provides an effective method to approximate the $Q(s, a)$, that is *deep Q-network*(DQN) [32,40,41].

3.2. Deep Q-Network for Top-Coal Caving

The framework of deep Q-network for top-coal caving is based on the work in [54]. Consider the state of top-coal caving system where $s_i, i = 1, 2, \dots, n$ are continuous variables and the actions $a_j, j = 1, 2, \dots, d$ are discrete variables. DQN employs a deep neural network to approach $Q(s, a)$ in Equation (2). The Q-network is formalized as $Q(s, a; \theta)$, where θ is the parameter of the neural network and it needs to be trained. Hence, there are two import issues for training θ : the samples to train the neural network and the loss function to indicate the effect of neural network training.

During the top coal caving, the decision system gets the state s from the environment in each step, and the decision mechanism gives an action a to the window. The window executes the action, and then the decision system gets the next state s' and calculates the reward R by the captured coal and rock. Hence, each step can produce a quaternion $\{s, a, R, s'\}$ that is a sample to train the Q-network. In the DQN, the experience store and replay are the key technologies to deal with the samples. For the top-coal caving, the samples of the quaternion $\{s, a, R, s'\}$ produced by each step are stored in the experience dataset, and then a batch of samples are selected randomly for replaying to train the parameters.

The loss function for training Q-network is based on Equation (5). Because dynamic programming [53] deduces the optimal decision from the destination inversely, the term $R(s, a) + \gamma \max_a Q(s', a)$ is the related value of next state; hence, this term is regarded as the target the current $Q(s, a)$ should approach. That target network is shown in Equation (6):

$$Q_{target}(s, a; \theta^-) = R(s, a) + \gamma \max_a Q(s', a; \theta^-) \quad (6)$$

Hence, the loss function is formalized as Equation (7).

$$f_{loss} = \left(Q(s', a; \theta) - R(s, a) - \gamma \max_a Q(s', a; \theta^-) \right)^2 \quad (7)$$

Training the parameters means making the loss function close to zero. In practice, the target network uses the same architecture of Q-network and the parameter θ^- is not trained but gets the Q-network parameter θ to update periodically. The Q-network is trained during the top-coal caving, at the beginning of the process, in order to cover as much of the state and action as possible. The window action is chosen by ϵ greedy algorithm [33], and the ϵ value is set favorably for exploration. The details of DQN for top-coal caving are shown as Algorithm 1.

Algorithm 1: DQN for top-coal caving.

```

1 Initialize  $N$ , the capacity of experience data set,  $n = 0$ , the number of stored experience
2 Initialize  $M$ , the episode for training; Initialize  $K$ , the maximum step of each episode
3 Initialize  $J$ , the windows number
4 Initialize  $\gamma, \alpha$  the parameters for target-network
5 Initialize  $C$ , the name of target-network updates its parameters from Q-network
6 Initialize  $\epsilon$ , the index to explore the relationship of state space and action space
7 Initialize parameter  $\theta$  of Q-network  $Q(s, a; \theta)$ , with constant
8 Initialize parameter  $\theta^-$  of target-network  $Q(s, a; \theta^-)$  with  $\theta^- = \theta$ 
9 for  $m \leftarrow 1$  to  $M$  do
10   Load the initial scenario top-coal caving.
11   for  $k \leftarrow 1$  to  $K$  do
12     for  $j \leftarrow 1$  to  $J$  do
13       Get state  $s_k$  With probability  $\epsilon$  select a random action  $a_k$  otherwise select
          $a_k = \arg \max_a Q(s, a; \theta)$ 
14       Window  $j$  executes  $a_k$ 
15       Observe reward  $R_k$  and next state  $s_{k+1}$ 
16       Store experience  $\{s_k, a_k, R_k, s_{k+1}\}$ ,  $n = n + 1$ 
17       if  $n > N$  then
18         Sample random minibatch of experiences  $\{s_t, a_t, R_t, s_{t+1}\}$  from experience
           storage
19         According to Equation (6), calculate the target
20         According to Equation (7), calculate the loss
21         Train the Q-net based on the loss by stochastic gradient descent [55]
22       end
23       if  $k \bmod C = 0$  then
24         Update target-network by  $\theta^- = \theta$ 
25       end
26     end
27   end
28 end

```

4. Experiment on Top-Coal Caving*4.1. DQN Model of Top-Coal Caving*

In the 3D simulation platform introduced in Section 2, the state space of each window is $\mathbb{S} = \{s_1, s_2\}$, s_1 is total particle number of windows checking area, and s_2 is the percentage of coal particle in s_1 . The action space of window is $\mathbb{A} = \{a_1, a_2\}$, and $a_1 = 0$, and $a_2 = 1$ mean close and open action, respectively. Hence, we establish the Q-network architecture as a four-layer fully connected neural network, and the parameters are set as Table 3. The reward R of each step depends on the captured particles formalized Equation (8).

$$R = n_c \times r_c + n_r \times r_r \quad (8)$$

where n_c and n_r are, respectively, the captured particles of coal and rock, and r_c and r_r are, respectively, the reward of capturing a coal and a rock. In this paper, $r_c = 1$ and $r_r = -3$.

Table 3. Architecture of DQN.

	Input Layer	Hidden Layer 1	Hidden Layer 2	Output Layer
Number of neurons	2	56	128	2
Initial θ	0.5	0.5	0.5	0.1

To distinguish which falling particle belong to which window, four clapboards are added to the deposition area, as shown in Figure 3a. The area for checking the state is the stereo region for A to B, as shown in Figure 3b. In the simulation, if the particle location is lower than the bottom of the tail-beam, that means it is captured by the window. The reward of each window is calculated by Equation (8).

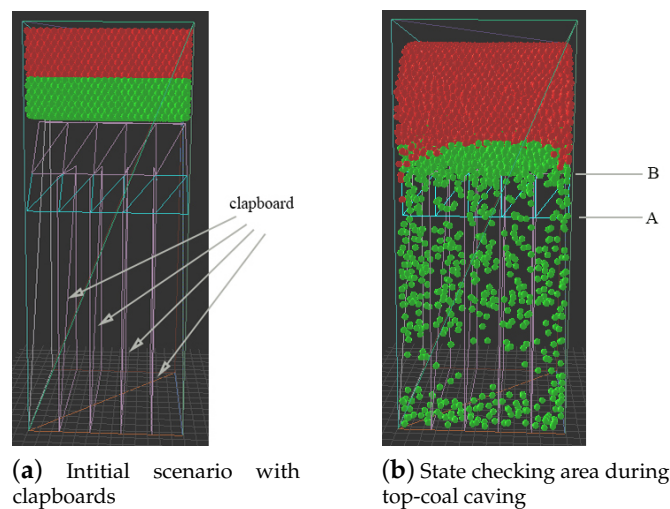


Figure 3. Setting for training DQN: (a) The clapboards are added between each hydraulic support for distinguishing the particles belongs to which windows, while they are removed in the testing. (b) The state check area is the stereo area between A and B of each hydraulic support; this area is the range of the window action.

4.2. Experiment and Result Analysis

Two algorithms were used to control window, the proposed DQN and the “close window if rocks emergency”, which are denoted by “DQN” and “Cmethod”, respectively. In the experiment, the thickness of coal and rock were set as 2 m, because the thickness was not the determining factor to validate the proposed algorithm. The radius of the particles was 0.15 m. The other parameters of training DQN were $batchsize = 100$, $N = 10000$, $C = 300$, $\alpha = 0.001$, $\gamma = 0.99$, and $\epsilon = 0.9$. For the DQN algorithm, the first step was to fill the experience dataset, and then train the Q-network. The experiences distribution shown in Figure 4 indicates that the states for the Q-network are closely related to the reward.

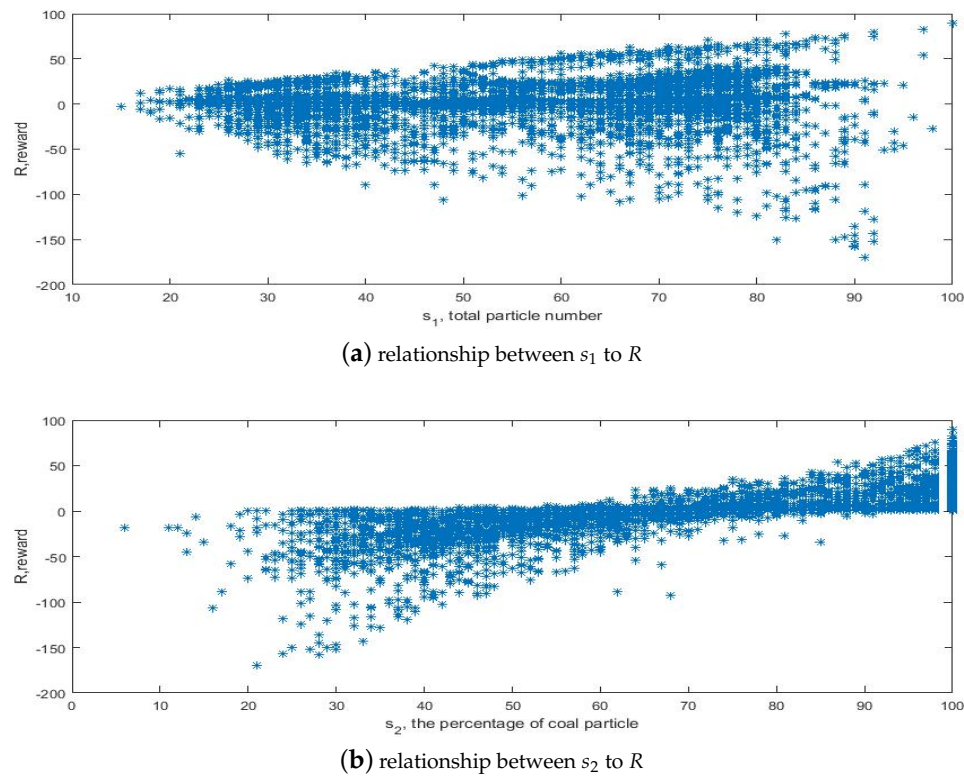


Figure 4. Experience distribution: (a) it is easy to find that the absolute value of R grows as s_1 increases; (b) the outline of R is proportional to s_1 . They indicate that a network could be found to approximate the relationship between particles number, coal rate, and reward.

The training processing is shown in Figure 5. The parameters of Q-network can converge to an optimal value obviously; hence, the optimal action-value function of $Q(s, a)$ can be approximated by a deep neural network.

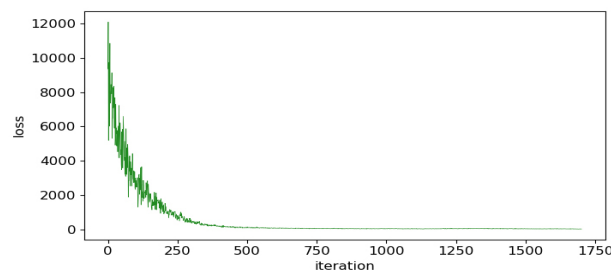


Figure 5. Training process. The loss convergences to a very small value after 500 training iterations.

After the parameters of Q-network were trained, 10 tests were carried out to validate the effectiveness of the proposed algorithm. The typical final scenario is shown in Figure 6. It is clear that, along the top of shield-beam, the coal is covered by rock, and the boundary is not a smooth line. Hence, in the next step, the hydraulic support moves forward; if the window control criterion were “close window if rock emergency”, the top-coal would be difficult to capture completely. DQN could avoid the problem since in Figure 6a the coal on the tail-beam is nearly captured completely. Furthermore, Figure 6b shows the boundary of the rock layer on the tail-beam obviously; the performance is the same as with 2D simulation.

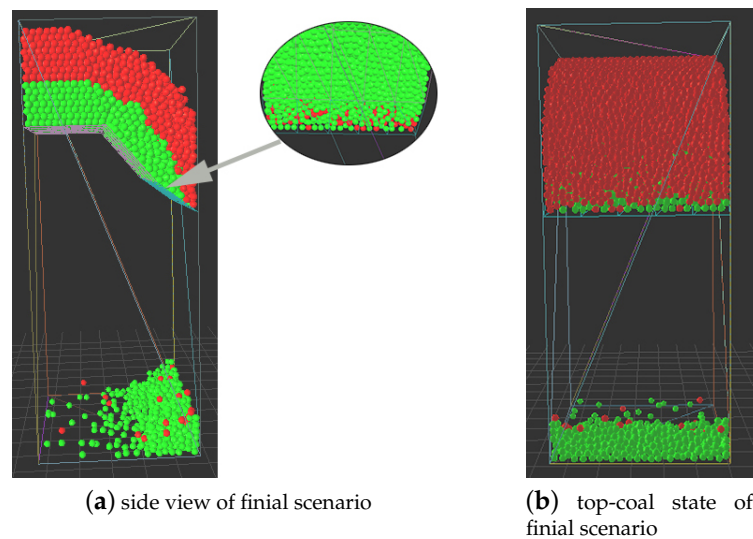


Figure 6. Final scenario: (a) side view, with the bottom view in the subgraph; and (b) the top-coal state.

The details of the experiment results are shown in Table 4. For the average value of each index, we can find that the captured coal particles and the reward of DQN are greater than for the Cmethod obviously with a high coal rate and low rock rate. That means the economic effect of top-coal by DQN would be better than by Cmethod.

Table 4. Experiment result.

No.	Coal Number		Rock Number		Coal Rate		Rock Rate		Reward	
	Cmethod	DQN	Cmethod	DQN	Cmethod	DQN	Cmethod	DQN	Cmethod	DQN
1	566	680	2	21	0.996	0.970	0.004	0.030	560	617
2	683	704	28	28	0.961	0.962	0.039	0.038	599	620
3	680	693	25	24	0.965	0.967	0.035	0.033	605	621
4	655	667	16	15	0.976	0.978	0.024	0.022	607	622
5	672	664	17	10	0.975	0.985	0.025	0.015	621	634
6	649	676	9	13	0.986	0.981	0.014	0.019	622	637
7	646	701	6	21	0.991	0.971	0.009	0.029	628	638
8	673	675	15	12	0.978	0.983	0.022	0.017	628	639
9	682	670	18	10	0.974	0.985	0.026	0.015	628	640
10	681	693	16	8	0.977	0.989	0.023	0.011	633	669
Average	658.7	682.3	15.2	16.2	0.978	0.977	0.022	0.023	613.1	633.7

5. Conclusions

This paper aims to get the optimal decision of the window's action of hydraulic support in top-coal caving by control theory. A DQN method based on the Markov optimal decision process is proposed according to the property of the top-coal caving. The effectiveness of the method was validated by the created 3D simulation platform. The experimental results show that:

- (1) The DQN method can get more coal particles obviously than the classical method with a very small price of increasing the rock rate. In the 10 tests, the average coal particles of the classical method and DQN are 658.7 and 682.3, respectively. Meanwhile, the rock rate of the DQN only rises 0.001.
- (2) The reward of the window's action by the DQN is better than the classical method. In the 10 tests, the average reward of the DQN is 633.7 while the classical method is 613.1. That means the DQN can produce more benefits than the classical method.

This paper tries to resolve the key problem of top-coal caving by artificial intelligent methodology, and the simulation results validate one of the ways that deep learning is applied to the top-coal caving. That means the optimal decision of windows would be regulated adaptively and intelligently. Even though the simulation result is better than that of the classical method, there are several issues that should be researched deeply in future work.

- (1) The state of the DQN is selected as the total number of the particles and the coal rate. At present, our method is only used in the simulation; one of the obstacles for practice application is that the DQN needs the data of the state. However, the data are difficult to obtain in practice; hence, in future work, we will try to use a deep neural network to approximate the needed data from other geological information.
- (2) The DQN gets the optimal Q-value by training, therefore there should be as much experience as possible, while, in practice, the experience obtained from top-coal caving is not as convenient as simulation. Hence, in future work, the learning mechanism of the DQN will be researched to get a lightweight learning framework based on the state space.

At present, the top-coal caving is mainly applied in China, and the serviceability of this method would be limited to Chinese coal mines. If the above issue is resolved, this method could be applied in practice. It could obviously increase the benefit of the top-coal caving and decrease the number of workers. Because the issue of the optimal decision for the control equipment in the other mineral resources is similar, the proposed method could be useful for other equipment in coal mining, and even for the optimal decision of the equipment in the underground metal mine.

Author Contributions: Algorithm framework, Y.Y.; Code and validation, X.L.; and 3D simulation platform of top-coal caving, H.L., D.L., and R.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by The National Key Research and Development Program of China (No. 2018YFC0604500), Henan Province Scientific and Technological Project of China (Nos. 192102210100 and 172102210270), and Key Scientific and Research Projects of Universities in Henan province of China (Nos. 19A413008 and 17A480007).

Acknowledgments: The authors are grateful to the staff at the Tashan coal mine for their assistance for establishing the 3D simulation. The authors express their sincere thanks to the editor and the anonymous reviewers for their constructive suggestions, which helped improve the manuscript.

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

References

1. Leonard, M.D.; Michaelides, E.E.; Michaelides, D.N. Substitution of coal power plants with renewable energy sources—Shift of the power demand and energy storage. *Energy Convers. Manag.* **2018**, *164*, 27–35. [\[CrossRef\]](#)
2. Khatib, H. IEA World Energy Outlook 2010—A comment. *Energy Policy* **2011**, *39*, 2507–2511. [\[CrossRef\]](#)
3. Xu, G.; Wang, W. China's energy consumption in construction and building sectors: An outlook to 2100. *Energy* **2020**, *195*, 117045. [\[CrossRef\]](#)
4. Energetika. *BP Energy Outlook: 2019 Edition*; BP Press: London, UK, 2019.
5. Mohanta, S.; Mishra, B.; Biswal, S. An emphasis on optimum fuel production for Indian coal preparation plants treating multiple coal sources. *Fuel* **2010**, *89*, 775–781. [\[CrossRef\]](#)
6. Jingchao, Z.; Kotani, K.; Saijo, T. Low-quality or high-quality coal? Household energy choice in rural Beijing. *Energy Econ.* **2019**, *78*, 81–90. [\[CrossRef\]](#)
7. Eremin, M.; Esterhuizen, G.; Smolin, I. Numerical simulation of roof cavings in several Kuzbass mines using finite-difference continuum damage mechanics approach. *Int. J. Min. Sci. Technol.* **2020**. [\[CrossRef\]](#)
8. Dobson, J.A.; Riddiford-Harland, D.L.; Bell, A.F.; Wegener, C.; Steele, J.R. Effect of shaft stiffness and sole flexibility on perceived comfort and the plantar pressures generated when walking on a simulated underground coal mining surface. *Appl. Ergon.* **2020**, *84*, 103024. [\[CrossRef\]](#)
9. Vakili, A.; Hebblewhite, B. A new cavability assessment criterion for Longwall Top Coal Caving. *Int. J. Rock Mech. Min. Sci.* **2010**, *47*, 1317–1329. [\[CrossRef\]](#)

10. Alehossein, H.; Poulsen, B.A. Stress analysis of longwall top coal caving. *Int. J. Rock Mech. Min. Sci.* **2010**, *47*, 30–41. [[CrossRef](#)]
11. Si, G.; Jamnikar, S.; Lazar, J.; Shi, J.Q.; Durucan, S.; Korre, A.; Zavšek, S. Monitoring and modelling of gas dynamics in multi-level longwall top coal caving of ultra-thick coal seams, part I: Borehole measurements and a conceptual model for gas emission zones. *Int. J. Coal Geol.* **2015**, *144–145*, 98–110. [[CrossRef](#)]
12. Zhang, Q.; Yue, J.; Liu, C.; Feng, C.; Li, H. Study of automated top-coal caving in extra-thick coal seams using the continuum-discontinuum element method. *Int. J. Rock Mech. Min. Sci.* **2019**, *122*, 104033. [[CrossRef](#)]
13. Le, T.D.; Zhang, C.; Oh, J.; Mitra, R.; Hebblewhite, B. A new cavability assessment for Longwall Top Coal Caving from discontinuum numerical analysis. *Int. J. Rock Mech. Min. Sci.* **2019**, *115*, 11–20. [[CrossRef](#)]
14. Gu, Q.; Ru, W.; Tan, Y.; Ning, J.; Xu, Q. Mechanical Analysis of Weakly Cemented Roof of Gob-side Entry Retaining in Fully-Mechanized Top Coal Caving Mining. *Geotech. Geol. Eng.* **2019**, *37*, 2977–2984. [[CrossRef](#)]
15. Zhang, Q.; Yuan, R.; Wang, S.; Li, D.; Li, H.; Zhang, X. Optimizing Simulation and Analysis of Automated Top-Coal Drawing Technique in Extra-Thick Coal Seams. *Energies* **2020**, *13*, 232. [[CrossRef](#)]
16. Guo, W.; Tan, Y.; Bai, E. Top coal caving mining technique in thick coal seam beneath the earth dam. *Int. J. Min. Sci. Technol.* **2017**, *27*, 165–170. [[CrossRef](#)]
17. Basarir, H.; Oge, I.F.; Aydin, O. Prediction of the stresses around main and tail gates during top coal caving by 3D numerical analysis. *Int. J. Rock Mech. Min. Sci.* **2015**, *76*, 88–97. [[CrossRef](#)]
18. Xie, Y.S.; Zhao, Y.S. Numerical simulation of the top coal caving process using the discrete element method. *Int. J. Rock Mech. Min. Sci.* **2009**, *46*, 983–991. [[CrossRef](#)]
19. Song, Z.; Zhang, J. Numerical Simulation of Top-Coal Thickness Effect on the Top-Coal Recovery Ratio by Using DEM Method. *Electron. J. Geotech. Eng.* **2015**, *20*, 3795–3796.
20. Wang, J.; Zhang, J.; Li, Z. A new research system for caving mechanism analysis and its application to sublevel top-coal caving mining. *Int. J. Rock Mech. Min. Sci.* **2016**, *88*, 273–285. [[CrossRef](#)]
21. Liu, C.; Li, H.; Jiang, D. Numerical simulation study on the relationship between mining heights and shield resistance in longwall panel. *Int. J. Min. Sci. Technol.* **2017**, *27*, 293–297.
22. Shahani, N.M.; Wan, Z.; Guichen, L.; Siddiqui, F.I.; Pathan, A.G.; Yang, P.; Liu, S. Numerical analysis of top coal recovery ratio by using discrete element method. *Pak. J. Eng. Appl. Sci.* **2019**, *25*, 26–35.
23. Liu, C.; Li, H.; Ying, Z. Method of synergetic multi-windows caving in longwall top coal caving working face. *J. China Coal Soc.* **2019**, *44*, 2632–2640.
24. Feng, G.; Wang, P. Simulation of recovery of upper remnant coal pillar while mining the ultra-close lower panel using longwall top coal caving. *Int. J. Min. Sci.* **2020**, *30*, 55–61. [[CrossRef](#)]
25. Le, T.D.; Mitra, R.; Oh, J.; Hebblewhite, B. A review of cavability evaluation in longwall top coal caving. *Int. J. Min. Sci. Technol.* **2017**, *27*, 907–915. [[CrossRef](#)]
26. Zhang, N.; Liu, C.; Wu, X.; Ren, T. Dynamic random arching in the flow field of top-coal caving mining. *Energies* **2018**, *11*, 1106. [[CrossRef](#)]
27. Unver, B.; Yasitli, N. Modelling of strata movement with a special reference to caving mechanism in thick seam coal mining. *Int. J. Coal Geol.* **2006**, *66*, 227–252. [[CrossRef](#)]
28. Nikitenko, M.; Kizilov, S.; Nikolaev, P.; Kuznetsov, I. *Technical Devices of Powered Roof Support for the Top Coal Caving as Automation Objects*; IOP Conference Series: Materials Science and Engineering; IOP Publishing: Bristol, UK, 2018; Volume 354, p. 012014.
29. Khanal, M.; Adhikary, D.; Balusu, R. Evaluation of mine scale longwall top coal caving parameters using continuum analysis. *Min. Sci. Technol.* **2011**, *21*, 787–796. [[CrossRef](#)]
30. Li, Z.; Xu, J.; Yu, S.; Ju, J.; Xu, J. Mechanism and prevention of a chock support failure in the longwall top-coal caving faces: A case study in Datong coalfield, China. *Energies* **2018**, *11*, 288. [[CrossRef](#)]
31. Cui, F.; Dong, S.; Lai, X.; Chen, J.; Cao, J.; Shan, P. Study on Rule of Overburden Failure and Rock Burst Hazard under Repeated Mining in Fully Mechanized Top-Coal Caving Face with Hard Roof. *Energies* **2019**, *12*, 4780. [[CrossRef](#)]
32. Yates, C.A.; Ford, M.J.; Mort, R.L. A multi-stage representation of cell proliferation as a Markov process. *Bull. Math. Biol.* **2017**, *79*, 2905–2928. [[CrossRef](#)]
33. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*; MIT Press: Cambridge, MA, USA, 2018.
34. Luo, B.; Liu, D.; Huang, T.; Wang, D. Model-free optimal tracking control via critic-only Q-learning. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *27*, 2134–2144. [[CrossRef](#)] [[PubMed](#)]

35. Rummery, G.A.; Niranjan, M. *On-Line Q-Learning Using Connectionist Systems*; University of Cambridge, Department of Engineering: Cambridge, UK, 1994; Volume 37.
36. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436. [[CrossRef](#)] [[PubMed](#)]
37. Seide, F.; Li, G.; Yu, D. Conversational speech transcription using context-dependent deep neural networks. In Proceedings of the Twelfth Annual Conference of the International Speech Communication Association, Florence, Italy, 27–31 August 2011; pp. 437–440.
38. Sainath, T.N.; Mohamed, A.R.; Kingsbury, B.; Ramabhadran, B. Deep convolutional neural networks for LVCSR. In Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 26–31 May 2013; pp. 8614–8618.
39. Schütt, K.; Gastegger, M.; Tkatchenko, A.; Müller, K.R.; Maurer, R.J. Unifying machine learning and quantum chemistry with a deep neural network for molecular wavefunctions. *Nat. Commun.* **2019**, *10*, 1–10. [[CrossRef](#)] [[PubMed](#)]
40. Clary, K.; Tosch, E.; Foley, J.; Jensen, D. Let's Play Again: Variability of Deep Reinforcement Learning Agents in Atari Environments. In Proceedings of the Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, Montréal, QC, Canada, 3–8 December 2018.
41. Van Hasselt, H.; Guez, A.; Silver, D. Deep reinforcement learning with double q-learning. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, Phoenix, AZ, USA, 12–17 February 2016; pp. 2094–2100.
42. Wang, Z.; Schaul, T.; Hessel, M.; Van Hasselt, H.; Lanctot, M.; De Freitas, N. Dueling network architectures for deep reinforcement learning. *arXiv* **2015**, arXiv:1511.06581. Available online: <https://arxiv.org/pdf/1511.06581.pdf> (accessed on 1 July 2019).
43. Schaul, T.; Quan, J.; Antonoglou, I.; Silver, D. Prioritized experience replay. *arXiv* **2015**, arXiv:1511.05952. Available online: <https://arxiv.org/pdf/1511.05952.pdf> (accessed on 1 July 2019).
44. Hoel, C.J.; Driggs-Campbell, K.; Wolff, K.; Laine, L.; Kochenderfer, M.J. Combining Planning and Deep Reinforcement Learning in Tactical Decision Making for Autonomous Driving. *IEEE Trans. Intell. Veh.* **2019**, *1*, 1. [[CrossRef](#)]
45. Kalashnikov, D.; Irpan, A.; Pastor, P.; Ibarz, J.; Herzog, A.; Jang, E.; Quillen, D.; Holly, E.; Kalakrishnan, M.; Vanhoucke, V.; et al. Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation. In Proceedings of the 2nd Conference on Robot Learning (CoRL 2018), Zurich, Switzerland, 29–31 October 2018.
46. Hessel, M.; Soyer, H.; Espenholt, L.; Czarnecki, W.; Schmitt, S.; van Hasselt, H. Multi-task deep reinforcement learning with popart. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 3796–3803.
47. Palmer, G.; Tuyls, K.; Bloembergen, D.; Savani, R. Lenient multi-agent deep reinforcement learning. In Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems. International Foundation for Autonomous Agents and Multiagent Systems, Stockholm, Sweden, 10–15 July 2018; pp. 443–451.
48. Šmilauer, V.; Ning, G.; Alexander, E.; Bruno, C.; Raphael, M.; Thomas, S.; Francois, K.; Luc, S.; Emanuele, C.; Sergei, D.; et al. *Yade Documentation*, 2nd ed.; The Yade Project, Grenoble University: Grenoble, France, 2015.
49. Šmilauer, V.; Ning, G.; Alexander, E.; Bruno, C.; Raphael, M.; Thomas, S.; Francois, K.; Luc, S.; Emanuele, C.; Sergei, D.; et al. Using and Programming. In *Yade Documentation*, 2nd ed.; The Yade Project, Grenoble University: Grenoble, France, 2015.
50. Šmilauer, V.; Ning, G.; Alexander, E.; Bruno, C.; Raphael, M.; Thomas, S.; Francois, K.; Luc, S.; Emanuele, C.; Sergei, D.; et al. Reference Manual. In *Yade Documentation*, 2nd ed.; The Yade Project, Grenoble University: Grenoble, France, 2015.
51. Li, Q.; Yang, Y.; Li, H.; Fei, S. Intelligent control strategy for top coal caving based on Q-learning model. *Ind. Mine Autom.* **2020**, *46*, 72–79. (In Chinese) [[CrossRef](#)]
52. Šmilauer, V.; Chareyre, B. DEM formulation. In *Yade Documentation*, 2nd ed.; The Yade Project, Grenoble University: Grenoble, France, 2015.
53. Bellman, R. Dynamic programming. *Science* **1966**, *153*, 34–37. [[CrossRef](#)]

54. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529. [[CrossRef](#)]
55. Bottou, L. Large-scale machine learning with stochastic gradient descent. In Proceedings of the 19th International Conference on Computational Statistics, Paris, France, 22–27 August 2010; Springer: Berlin, Germany, 2010; pp. 177–186.



© 2020 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).