Intelligent Agent-Based Simulation and Deep Reinforcement Learning for Smart Mining Operations

Solomon Acheampong, Pedro Pablo Vasquez-Coronado and Eugene Ben-Awuah
Mining Optimization Laboratory (MOL)
Laurentian University, Sudbury, Canada

ABSTRACT

Real-time intelligent decision-making in a mining environment with huge amounts of mine operations data is the basis for smart mining. By leveraging a deep reinforcement learning algorithm using the Deep Q Networks (DQN), the objective of this research is to enhance shovel-truck productivity by accurately detecting and overcoming operational obstacles within the mining environment. The research aims to develop an intelligent system that effectively supervises the operations of a short-term open pit mine through the combination of simulation and intelligent agents. This includes modelling the mining operation and employing agent-based simulation to control equipment fleet and implementing a reinforcement learning agent for intelligent decision-making. The proposed methodology establishes a virtual experimental environment by incorporating a decision model supported by discrete event simulation. The intelligent supervisory agent is trained using deep reinforcement learning techniques to determine mining zones based on key performance indicators such as ore grade and tonnage. The model's effectiveness and performance are evaluated through the examination of experimental scenarios; utilizing experimental designs to verify and validate the proposed approach.

1. Introduction

Smart mining is a concept that involves the use of advanced technologies to optimize mining operations. One of the key components of smart mining is real-time intelligent decision-making, which requires the processing of huge amounts of mine operations data. To do this, researchers are looking into using deep reinforcement learning algorithms based on Deep Q Networks (DQN) to improve shovel-truck productivity by accurately detecting and getting around operational obstacles in a mining environment [44].

Simulation modelling is a powerful tool that has been widely used in various industries to replicate real-world scenarios and test different strategies and decision-making processes [26]. In the context of mining operations, simulation models can provide valuable insights into the performance of different mining strategies and help identify areas for improvement. On the other hand, deep reinforcement learning is a branch of artificial intelligence that combines deep learning and reinforcement learning techniques to enable intelligent agents to make optimal decisions in complex environments [25].

One of the key challenges in the mining industry is the need for real-time decision-making. Mining operations are complex and dynamic, and decisions need to be made quickly to avoid delays and maximize productivity. The use of intelligent systems that can make real-time decisions based on data analysis is therefore critical for the success of mining operations. The development of intelligent

ISBN: 978-1-55195-503-2

and autonomous mining systems is an area of active research, with many companies and research institutions investing in this field. The use of AI and simulation modelling is expected to play a key role in the development of these systems, enabling mining companies to optimize their operations and improve safety and efficiency.

In this article, we will explore how this approach can be used to optimize mining operations and improve efficiency and safety in the mining industry. The methodology involves the development of a simulation model that replicates mining operations and the application of deep reinforcement learning techniques to train an intelligent agent. The results of the experiments conducted using the simulation model demonstrate the effectiveness of the intelligent system in achieving the desired production targets and optimizing the allocation of resources.

By making a deep reinforcement learning-based model of an intelligent system, this research contributes to the field of simulation optimization of intelligent and autonomous mining systems. The findings of this research have implications for the mining industry, as they provide insights into the potential benefits of using intelligent systems for decision-making in mining operations.

2. Literature Review

The literature review covers three main areas: operational planning, simulation modelling, and artificial intelligence (AI), including machine learning (ML) and deep learning (DL).

2.1. Operational planning

Operational planning in short-term open pit mining refers to the process of developing and implementing a plan for the day-to-day operations of continuous mining. It involves several activities, including setting production targets, allocating resources, establishing schedules, and managing operations. The goal of operational planning is to ensure that the mining operates efficiently and effectively to achieve its strategic objectives.

The operational planning process typically begins with the development of a mining plan, which outlines the mining objectives, resource requirements, and production targets. This plan is then used to develop a detailed operational plan, which outlines the specific steps that will be taken to achieve these goals. This might include identifying the equipment needed, establishing schedules and workflows, and allocating resources to different parts of the mining operation.

This also involves monitoring and controlling the mining operation to ensure that it is meeting its target and operating effectively. These include the use of real-time data and performance metrics, as well as regular assessments of the mining operation to identify areas for improvement.

2.2. Simulation Modelling

Many industries use simulation on a regular basis. Its main objective is to mathematically or virtually express operations or concepts in real or hypothetical situations. The 1940s saw the beginning of simulation [5]. As a digital tool for resolving challenging real-world problems, simulation has developed over time to model neural cells in the brain. Simulation can be done in a variety of ways, including continuous, discrete, stochastic, and deterministic.

Simulation is the representation of real-world situations mathematically or visually in a virtual setting. It enables the realistic representation of arbitrary decisions made in real-world circumstances [38]. In recent decades, data analysis and calculation speed have increased, making simulation more accurate, practical, and realistic.

Hashemi and Sattarvand [17] used Discrete event simulation (DES) to evaluate the Sungun copper mine haulage system. The authors were able to construct intricate systems and truck-to-loader assignments using this method to track performance. When comparing the flexible truck assignment for loaders to the fixed assignment system, the authors found a 7.8% improvement.

Manriquez et al. [27] conducted a case study on a truck and shovel excavator mining system in snowfall conditions using discrete event simulation (DES). They found that a specific operating policy could increase productivity by 32% when considering previous snowfall events. DES's ability to simulate intricate mining systems and snow removal operations highlights its importance in enhancing efficiency.

A simulation model aims to simulate the behavior or operation of a modelled system to determine optimal settings and levels of operation for optimal performance. Performance is a key performance indicator (KPI), and parameters are equivalent to decision variables [33]. These elements are multi-attribute behaviors in supply chain decision-making. Intelligent agents communicate with the simulation to improve results.

Modern simulation modelling was examined by Jovanoski et al. [22] and Grigoryev [16] using discrete event, agent-based, and system dynamics techniques. Different levels of abstraction and particular applications are used with each method. Fig. 1 illustrates how each technique serves a particular range of abstraction levels and their associated applications. High-level strategic modelling is done with the help of system dynamics (SD). While agent-based simulation (AB) can range from highly abstract models where agents even represent businesses to highly detailed models where agents represent physical objects, discrete event (DE) modelling only allows for low and medium-level abstractions.

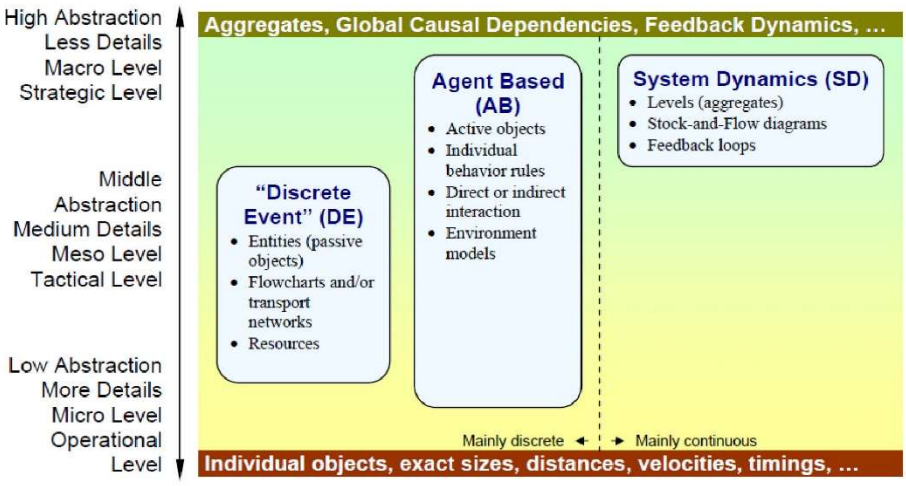

Fig. 1. Method in simulation modelling [22].

2.3. Artificial Intelligence

The debate on machine intelligence dates back to the 1940s, with Sir Alan Turing's paper [39] measuring machines' ability to mimic human intelligence. Arthur Samuel [32] introduces the theory of machines learning from raw data, categorized into two approaches: search, pattern recognition, learning, planning, and induction. DL is a subset of ML that uses neural networks and is inspired by McCulloch and Pitts' theory [28]. DL's feature extraction technique frees developers from manually crafting feature engineering algorithms.

Deep reinforcement learning (DRL) is a branch of reinforcement learning (RL) that incorporates the use of deep neural networks that mirror the functionality of the human brain. When compared to supervised and unsupervised machine learning techniques, RL enhances itself through rewards and penalties [35]. Deep Q network (DQN) is a DRL algorithm usually implemented on a Markov decision process stage to train agents to take decisive and good actions to maximize a reward function. The policy trains and improves itself through experience by performing a series of actions and receiving rewards (negative or positive) from the environment.

The Markov Decision Process (MDP) is a mathematical framework for simulating a decision-control scenario in a discrete-time stochastic environment [6]. With the actor acting as the agent, it consists of states and actions. The agent chooses an action at random among the possible actions from a particular state (or states) for each step in the process, which can exist in any random state. The environment motivates the agent by giving it a reward or feedback, Ra (s, s') in a looped process until an optimal policy is obtained. Fig. 2 presents the Markov decision process workflow.

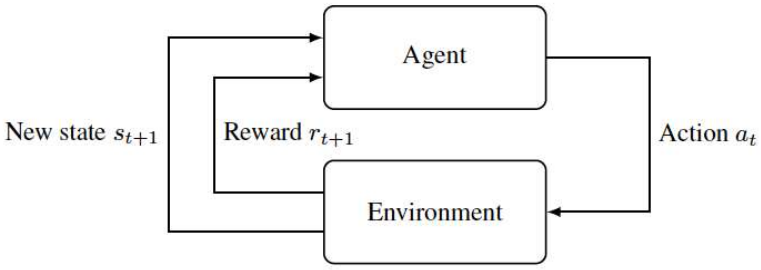

Fig. 2. Markov decision process workflow of agent using DQN [15].

Every policy employing the MDP framework must adhere to the four MDP elements. State (s), Action (a), Probability of action (Pa), also known as the Transition Probability Distribution Function, and Reward Function (Ra) are the terms used to characterize them.

The state of an instance determines its mode or configuration. Any random variable from a state space that a policy can be at any given time (t) is the state (s). Depending on the kind of system being modelled, the state space might either be continuous or discrete. All the orders and actions that may be carried out by a policy during a specified number of epochs or episodes are referred to as the Action (a).

2.4. Transition Probability Distribution Function

This is the probability that results in a shift in states (S). A state space, a transitional probability matrix that describes every possible transition, and an initial state (S1) from the state space make up the process. The matrix of transitions is shown in Fig. 3 below.

$$P = \begin{bmatrix} P_{1,1} & P_{1,2} & \dots & P_{1,j} & \dots & P_{1,S} \\ P_{2,1} & P_{2,2} & \dots & P_{2,j} & \dots & P_{2,S} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ P_{i,1} & P_{i,2} & \dots & P_{i,j} & \dots & P_{i,S} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ P_{S,1} & P_{S,2} & \dots & P_{S,j} & \dots & P_{S,S} \end{bmatrix}.$$

Fig. 3. Transitional probability matrix / The Markov Matrix [12].

The equation for the transition probability distribution function is shown in Eq. **Error! Reference source not found.** .

$$P_{ss'}^{a} = P[S_t + 1 = S' | S_t = S_1, a_t = a] \quad (1)$$

$P_{ss'}^{a}$ = Probability of action a, issued in state s, ending up in state s^1 with reward r

2.5. Reward Function Theory

After an agent takes a step and switches from one state s_1, to another state s_2, the environment responds with a reward (R_t), which is information. Maximizing total rewards is the agent's aim.

$$R_t = r_t + 1 + \gamma r_t + 2 + \gamma^2 r_t + 3 \dots \gamma^n r_t + n \quad (2)$$

From Eq. **Error! Reference source not found.**, γ is the discount factor between 0 and 1. Discount factors are the additives that enable the agent to comprehend the significance of future rewards. When the discount factor is set between 0 and 1, the agent starts to look for current and potential rewards accountability. The following illustrates the equation of the transition reward function.

$$R_{ss^1}^{a} = \mathrm{E}[r_t + 1 \mid S_t = S, a_t = a, S_t + 1 = S^1] \quad (3)$$

2.6. Bellman optimality

The Bellman optimality or equation, is used to calculate the expected value of a given policy π. From Eq. **Error! Reference source not found.**, a discount factor is introduced for the agent to account for immediate and future rewards. According to the Bellman equation [43], a long-term reward R_{t+n} for a given action is equal to the sum of the reward of the current action R_t and the expected reward from future actions R_{t+1}.

$$V(s) = \max_{\pi}(R(s,a) + \gamma V(s')) \quad (4)$$

From Eq. (4), V(s) is the value of the state. i.e., the numeric representation of a state, which guides the agent to find its path.

2.7. Policy Function

A Policy π is a function in charge of causing the agent to take an action (a) in a state (s). The sum of all the possible actions equals 1, i.e.

$$\Sigma_a \pi(s,a) = 1 \quad (5)$$

2.8. State Value Function (V^{π})

The expected total rewards to be received starting with a state is presented in Fig. 4.

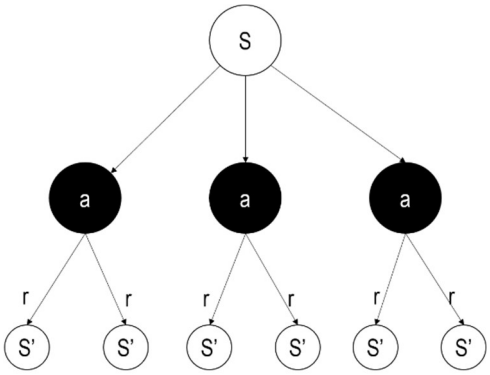

Fig. 4. Diagram illustrating state value function [12].

The state value function's mathematical model is presented in Eq. (6).

$$V^{\pi}(s) = \sum_{\alpha} \pi(s,a) \sum_{s'} P_{ss'}^{\alpha} R_{ss'}^{a} + \lambda \sum_{\alpha} \pi(s,a) \sum_{s'} P_{ss'}^{a} V^{\pi}(s') \quad (6)$$

From Eq. (6), $\pi(s,a)$ is the policy given state and action, $\sum_{s'} P_{ss'}^{\alpha} R_{ss'}^{a}$ is the sum product of the probability of action *a* issued in state s ending up in a new state s', returning an expected reward R.

2.9. Action Valueunction (Q^{π})

The action value is the worth of a decision made in a situation where the expected total reward is returned. It is depicted in Fig. 5.

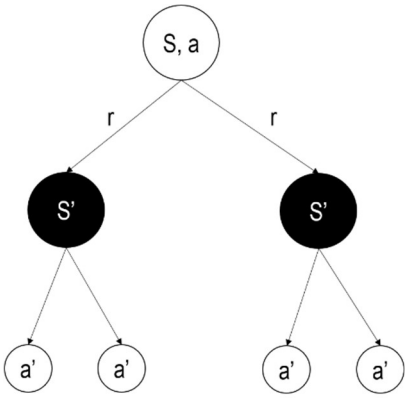

Fig. 5. Diagram illustrating action value function [12].

Mathematical model behind action value function is illustrated in Eq. (7).

$$Q^{\pi}(s,a) = \sum_{s'} P^{a}_{ss'} R^{a}_{ss'} + \gamma \sum_{s'} P^{a}_{ss'} V^{\pi}(s') \tag{7}$$

From Eq. (7), $Q^{\pi}(s,a)$ is the policy given state and action, $\sum_{s'} P^{\alpha}_{ss'} R^{a}_{ss'}$ is the sum product of the probability of action *a* issued in state s ending up in a new state s', returning an expected reward R given a state value of $V^{\pi}(s')$.

2.10. Optimal Policy

An optimal policy strategy is an action that maximizes some objective or reward in a system or process. The optimal policy maps states to actions, indicating the action that should be taken in each state to maximize the expected return. The optimal policy estimates the expected return of taking an action in a state. The optimal policy can be deduced as shown in Eq. (8).

$$V\pi(s) = \max_{\pi} V_{\pi}(s) \tag{8}$$

2.11. Greedy Policy

This function allows the agent to prioritize the most optimal steps. The relationship between state value and action value can be represented by Eq. (9).

$$V(s_t) = \max_{\pi} Q(s_t, a) \tag{9}$$

From Eq. (9), the Bellman equation can be expressed as shown in Eq. (10).

$$Q(s_t a_t) = r(s_t, a_t) + \gamma \frac{\max}{a_{t+1}} Q(s_{t+1}, a_{t+1}) \tag{10}$$

2.12. Deep Q Network

Deep learning is a type of machine learning approach that comprises two or more layers, weights and biases, and artificial neurons known as perceptron. Weights and biases play a very vital role in deep learning. The weight determines how much influence the input vector has on the output [43].

On the other hand, the biases, which are typically independent and constant with a value of 1, are added to the following layer. It serves as an assurance that even when an input vector contains zero values, there will still be an activation in the neurons [43]. Eq. (11) represent the mathematical equation of a neuron.

$$Y = \sum (weight * input) + bias \tag{11}$$

2.13. Stochastic Gradient Descent

When training a deep neural net on a dataset, the gradient descent functions as an optimization process to minimize the loss, normally called the loss function. The loss function serves as a check as to how the model performs given the parameters of weights and biases. The gradient descent helps in finding the precise parameter to use in order to achieve a good model [43].

2.14. Back-Propagation

Back propagation simply means calculating the gradient of a loss function. During the training process of a neural network, each weight within the network is calculated automatically with a deferential algorithm, and the calculated gradients are then used to update the weights [8].

2.15. Learning Rate

During the training process of a neural network, the learning rate determines the step size for which the error gradients are calculated [42]. The learning rate ranges from 0 to 1. The higher the rate, the more the network will overshoot the minimum. The lower the range, the longer the training process.

2.16. Deep Q Learning Network

DQN is a particular kind of reinforcement learning technique that is typically used on a MDP stage to teach agents to make strategic decisions to maximize a predetermined reward function. The policy engages in a series of actions and receives rewards from the environment, either negatively or positively, as it learns from experience. [9]. The decision-process workflow is shown in Fig. 6.

The Deep Q Network is primarily made up of the following techniques: Experience Replay is an approach used in the network to stabilize network updates. A group of transitions known as the replay buffer are added to the buffer at each data collection. Experience replay's entire purpose is to calculate loss and its gradient from a small batch of transitions taken from the replay buffer [13].

Target Network: The target network serves as an error measurement. The target network ensures that the agent replays each action it takes in an environment as if it were starting over each time, taking into account every action the agent undertakes there [10].

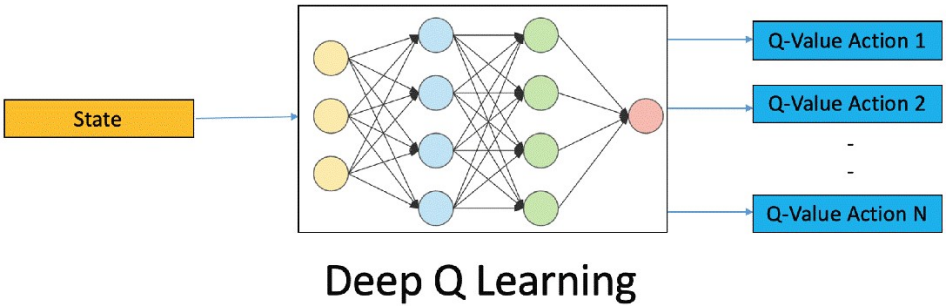

Fig. 6. Graphical examples of Deep Q learning [10].

Error! Reference source not found.illustrates the pseudocode of a DQN. Q (s, a) is for storing Q values for every time an action a_t is taken, usually by the policy or agent in a state *s*. The focus of the agent is to improve the Q value function. It learns from the Q value, which is an expected total return

from a given state-action pair. Anytime the agent makes a move, a reward is returned. Thus, R (s, a, s') becomes the new target the agent pursues from Q (s, a). γ-max is a discount factor. Discount factors cause reward functions to lose more of their value as the agent keeps on acting on the environment over time. This causes the agent to seek more rewards.

Table 1. Sample DQN algorithm.

***Hyperparameters**: replay buffer capacity N, reward discount factor γ, delayed steps C for target action-value function update, _greedy factor _*
***Input**: empty replay buffer D, initial parameters θ of action-value function Q*
Initialize target action-value function ˆQ with parameter ˆ θ ← θ
for** episode = 0, 1, 2, ... **do
Initialize environment and get observation O0
Initialize sequence S0 = (O0) and preprocess sequence φ0 = φ(S0)
for** t = 0, 1, 2, . . . **do
With probability _ select a random action At, otherwise select At = $argmax_a$ Q(φ(St), a; θ)
Execute action At and observe Ot+1 and reward Rt(s, a, s')
If the episode has ended, set Dt = 1. Otherwise, set Dt = 0
Set St+1 = (St,At,Ot+1) and preprocess φt+1 = φ(St+1)
Store transition (φt,At,Rt,Dt, φt+1) in D
Sample random minibatch of transitions (φi,Ai,Ri,Di, φi) from D
If Di = 0, set Yi = Ri + γ maxa_ˆQ(φi, a;θ). Otherwise, set Yi = Ri
Perform a gradient descent step on (Yi − Q(φi,Ai ; θ))2 with respect to θ
Synchronize the target ˆQ every C steps
If the episode has ended, break the loop
end for
end for

3. Problem statement

Data generated in the mining industry has recently increased exponentially, making it very difficult to gain knowledge in real time about the mining operation. As tons of data are processed with high-speed computers and smart software, big data mining has become more efficient and requires less work. The complexity of decision-making in mining has increased over the years due to the nature of the mineral deposits and the need to increase productivity of the operation. Mining data from day-to-day operations from monitoring devices and sensors includes the location of a truck or shovel, processing rate, and quantity of material mined.

Experimenting with these mining production systems in real life is risky, costly, and can lead to wrong decisions. Due to the complexity of these operations, the use of decision-support modelling tools is necessary and useful for understanding interactions and improving system performance. One well-known system modelling technique is simulation, which mimics the operation of the real world on a computer. By mimicking the operation of a real system, the simulation generates a predictable historical data about the system, and its observation allows inferences to be made concerning the system's operation or performance characteristics, including unexpected outages that could affect the entire system. Fig. 7 illustrates the statement of the problem comprised of mining activities and operations. Data gathered on mining operations such as production, fleet management, and dispatching can be difficult to analyze in real time as large volumes of data are generated over time.

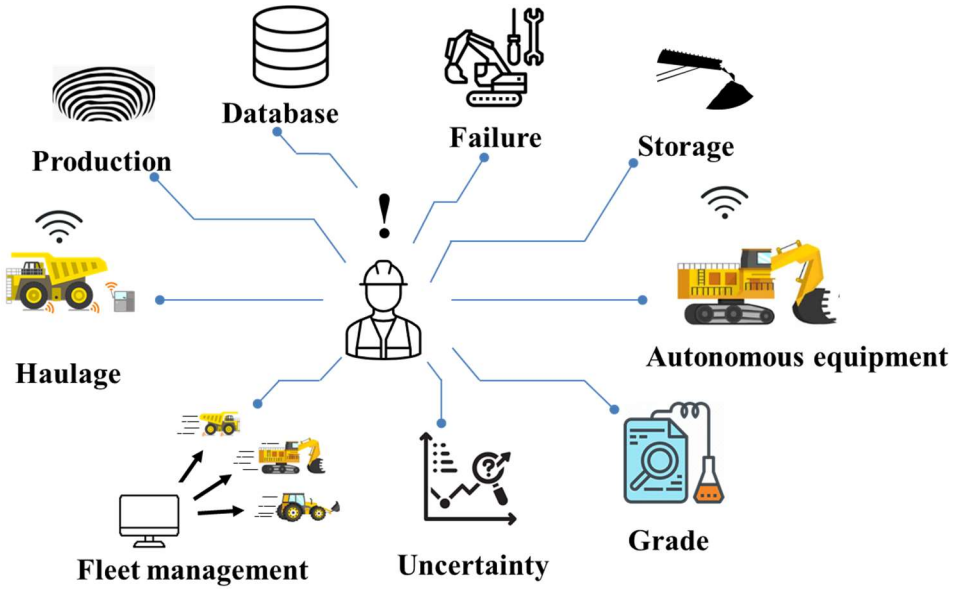

Fig. 8. Schematic representation of the problem statement

4. Methodology

The methodology outlined involves the utilization of discrete event simulation and an agent-based model, coupled with an intelligent supervisory agent, to establish a virtual environment for mining operational experiments. The research starts with long-term open pit production schedule data obtained from Whittle software optimization [36]. The schedule data contains three pushbacks, of which Pushback 1 is chosen as the focus of this research in a short-term mine planning context. The short-term production schedule data serves as an input to a simulation model that mimics the operation of the open pit mine, as shown in Fig. 9.

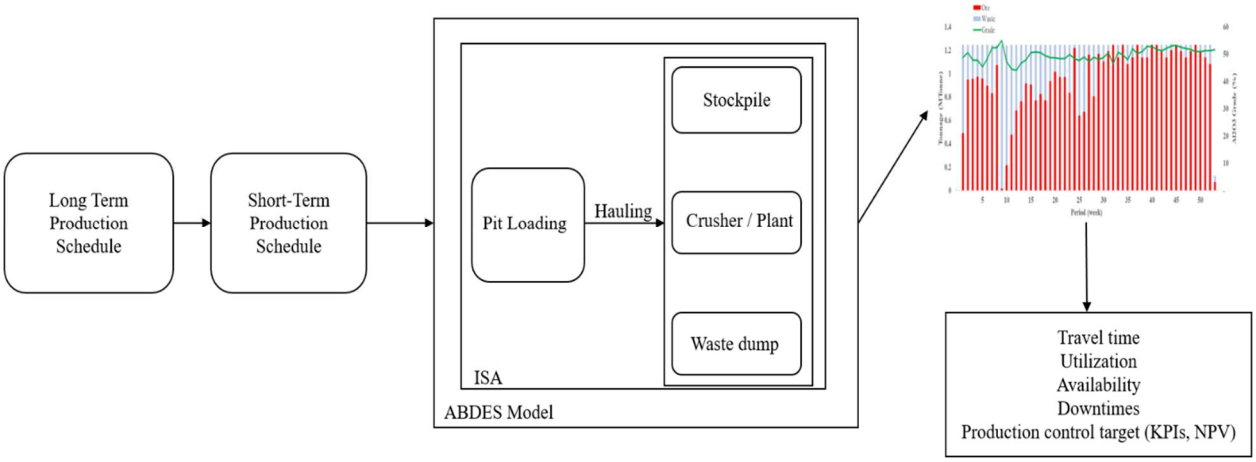

Fig. 10 Research methodology

The ISA is developed with reinforcement learning based on the DQN algorithm [11] and implemented with the object-oriented programming platform Java [12]. The DQN algorithm is suitable for both discrete state space and discrete action space. During implementation, the ISA is trained by monitoring the operational parameters, including ore tonnage and grade, truck cycle time, and shovel and plant availability. After training, the ISA is deployed to manage the mining operation by monitoring and allocating shovel digging areas as well as truck assignments to minimize production variations based on ore grade and ore tons. The ISA performs these operations by

observing the input states and parameters and choosing the best actions suitable for managing the continuous mining operation.

4.1. Proposed Algorithm

A pre-processing algorithm was created to divide the production schedule data into rows of mining blocks that represented a truck load of material before the simulation started. The algorithm generates mining zones, which function as locations for the allocation of loading equipment. The purpose of the mining zones created was to make it possible to monitor the ISA's performance. Fig. 11 illustrates the process flow integration of the ISA and the ABDES.

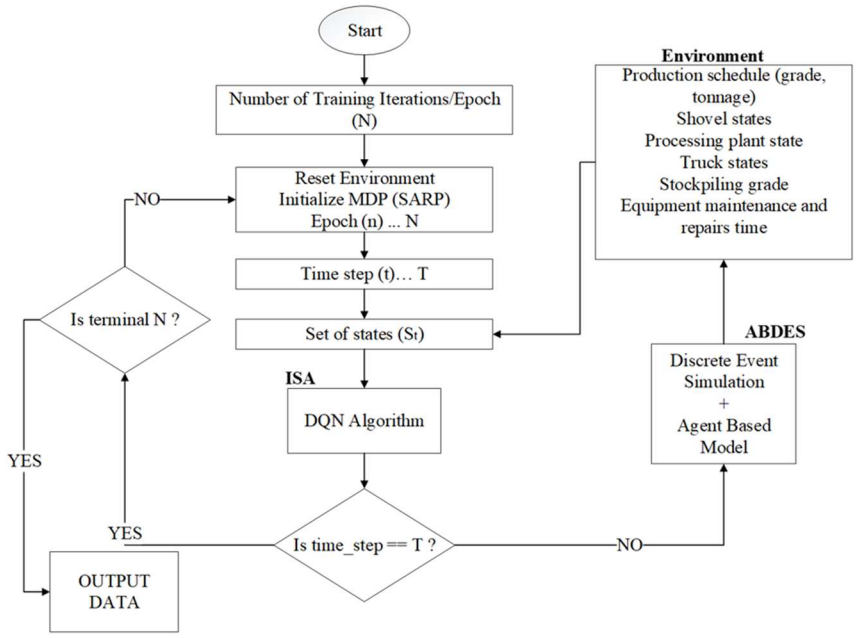

Fig. 11 Algorithm for Agent-Based Discrete Event Simulation and ISA integration

4.2. Agent-Based Discrete Event Simulation (ABDES)

This section covers the simulation sequence of the original design of the mining process. Fig. 12 illustrates the truck-shovel method of extracting earth materials from the pit.

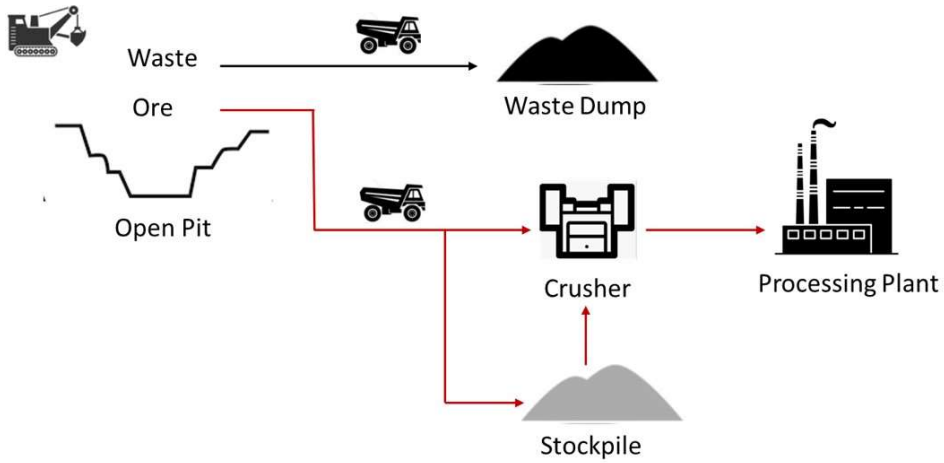

Fig. 12. Mining simulation model.

The method of simulation chosen for this research includes a DES, which simulates the flow of material from one point to the next. The ABM, on the other hand, models the behavior and activities of mine equipment and operations. Both are combined to form the ABDES, which is the interaction between the DES and ABM to provide a realistic mining operation environment.

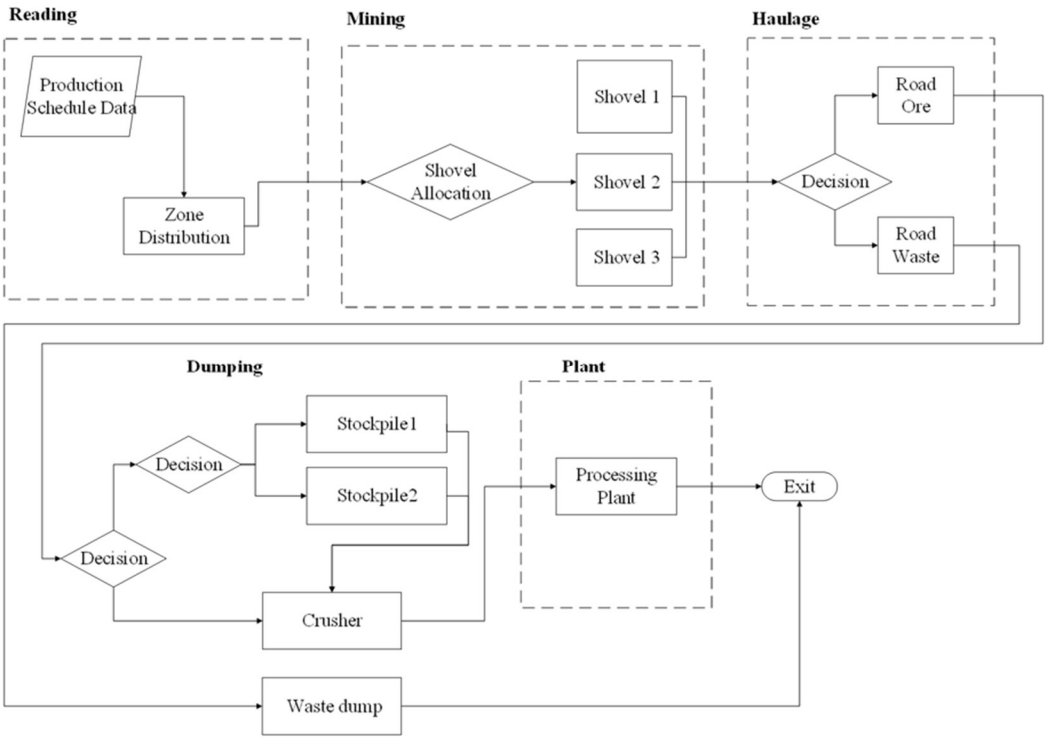

Fig. 13. Logical sequence for simulation model experiment.

4.3. Reading

Array matrix is used to store mining blocks from the production schedule data. This strategy speeds up computation and makes data accessible for the ISA to execute a tree search algorithm to find the best zones for mining. The DQN algorithm prioritizes high-grade blocks over low-grade blocks, separating high-grade, low-grade, and waste materials for mining to achieve targeted KPIs. The production model contains the following information in Table 1:

Table 1. Block model parameters.

Field	Description
BlockID	Block identification
Coordinate-X	Location in X
Coordinate-Y	Location in Y
Coordinate-Z	Location in Z
Rocktype:	0 Waste, 1 Ore
Tonnage	Amount of material
MetalContent	Contained metal
Recovery	Percentage of ore recovery
Al_2O_3 Grade	Aluminide grade
Al_2O_3 (tonnage)	Aluminide Tonnage

SiO_2 Grade	Silica grade
SiO_2 (tonnage)	Silica tonnage
Pushback	Mining phase
Period	Mining Period

4.4. Mining

The Mining node takes over the block entities from the Reading node. The Mining node deploys three shovels, and it is a requirement for the three shovels to work simultaneously. With the integration of the ISA, it ensures each shovel works in a different zone and continuously monitors the shovel activities. An agent-based model (ABM) represents shovel behavior, allowing monitoring of individual activities like loading, failure, maintenance, repairs, idle state, and active state. Trucks are modelled using ABM, with each truck acting as a self-decision sub-agent, enabling ISA monitoring of activities like idle, active, hauling, loading, unloading, failures, repairs, and maintenance.

4.5. Haulage

The Haulage node takes over truck entities from the Mining node, using 3D haul road profiles from Gems to design a haulage simulation.

4.6. Dumping

The Dumping node takes over the truck entities from the Haulage node. This node functions in the same way as in Scenario 1. However, one additional stockpile is added to segregate the quality of the stockpiled material using a predefined cut-off grade. Incoming ore materials are transported directly to the crusher if there is no queue. If there is a queue at the crusher, high-grade materials are transported to Stockpile 1, while low-grade materials are transported to Stockpile 2.

4.7. Plant

The Plant node takes over the material entities from the Dumping node. The plant is modelled using ABM to incorporate availability, utilization, maintenance, repair, failure, and downtime. Ore is sent continuously from the mine or stockpiles, and if not at full production capacity, material is received from the stockpile. Ore is reclaimed from either high- or low-grade stockpiles to minimize deviation.

4.8. Replication

The obtained values for the variables and parameters at the end of the simulation will differ significantly if the model is run more than once [14]. The objectives and necessary simulation accuracy should be used to justify the number of simulations for each individual test [19]. Within the probability distribution of all possible combinations, each replicated simulation represents a workable solution. According to Holford et al. [19], the number of simulation replicates should depend on the study's accuracy and goals.

A minimum range of 2 and a maximum of 10 replicates were established with which to run the simulation, and a 95% confidence interval was used to validate the number of replicates. Due to the different types of data and variations in the model, different numbers of replicates were used in each case.

Intelligent Supervisory Agent (ISA)

The proposed framework assumes that there are N total zones, each of which defines a set of shovels and trucks. It has been proposed that the shovels operate simultaneously, switching between various locations in accordance with real-world requirements. Shovels are given access to available trucks according to a schedule as soon as they are required. The model shows the various mining zones that the agent must select before beginning to mine. While the other zones are in the inferior level, the first zones are in the superior level. Zone distributions are viewed in Fig. 14.

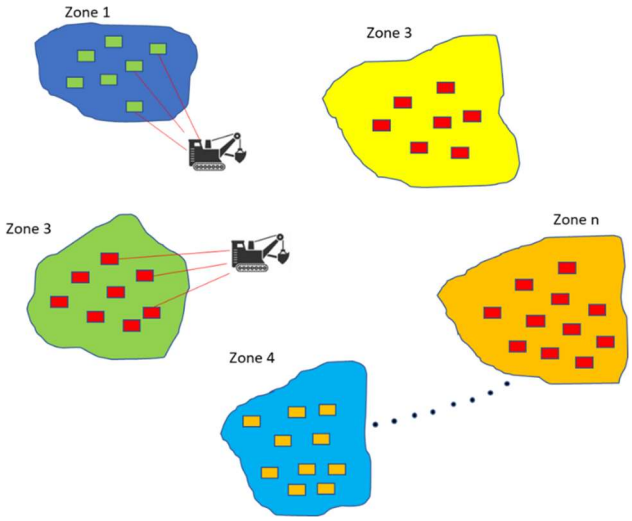

Fig. 14. Graphical representation of the zone's distribution.

4.9. Training implementation architecture

The grades and tonnages (rectangular boxes) contained in each of the zones are assumed to be known. However, these metrics are not exposed to the intelligent agent. The intelligent agent must uncover these values by training and learning from the environment. It first extracts the top-level zones and continues with the lower zones.

- Environment

The environment represents the problem to be solved. It can be described using a simulation. It is everything that comes after the agent's decision. The environment is modelled as states S, and each state s in S requires the agent to take specific actions. Failure to take the required action results in the agent receiving a negative reward or a penalty.

- State Space

The state space is a set of variables and all the possible values they can take. The State is an instance of the State Space. The State is the set of values a variable takes. At any period *t*, the state space is an observation of instances obtained from the environment. The observed states at each time step *t* are the dispatch system states, processing plant states, and production schedule states (tonnage and grade).

- Action

The environment makes available a set of actions from which the agent chooses an action. The agent influences the environment with these actions, and the environment may change states as a response to the agent's action. In this study, at each time step *t,* the agent is provided with four (4) actions with each action containing set of algorithms that control the state of the simulation. To ensure the simulation converges and the mine operation yields a high NPV, the agent is required to find which actions will support the target objective, such as:

0: Take no Action

1: Choose zones with good minimum deviation

2: Allocate available shovels for selected zones

3: Allocate available trucks for the active shovels

The agent uses two criteria to select zones: ore grade and ore tonnage. A target is set, and random restricted actions are made, starting from top-level zones. If the agent makes the wrong decision, penalty feedback is rewarded. Each block in the selected zone contains an available ore grade, which is subtracted from the target grade and tonnage per week. This minimum deviation reward is cumulative for blocks in each zone, ensuring the highest possible rewards for individual zones.

- Reward function

This function represents feedback on how well the last action contributed to achieving the target objective of the environment. The reward function $r_t(s',a)$ helps the ISA to learn the best possible actions to take to achieve its goals. The objective of the ISA is to maximize its rewards over time. Therefore, we introduce a discount factor (γ), which controls how the ISA will account for future rewards. The simulation is designed in such a way that, when the agent chooses the correct block sequencing, accurate dispatching, improved processing plant productivity, and reduced equipment maintenance and repairs, the ISA will receive a high score from the reward function at terminal stage T. The score is calculated based on the discounted cashflow accumulated over time, in weeks.

Fig. 15 represents the neural network structural design of the ISA. It comprises an input node, a hidden layer, and an output node. The input nodes are state observation snapshots from the environment. The hidden layer is where major computations occur. It takes an input from the previous node and performs a weighted sum (wx + b). The weighted sum is then transformed using a "relu" activation function, which introduces non-linearity to the model, allowing it to learn complex relationships in the data.

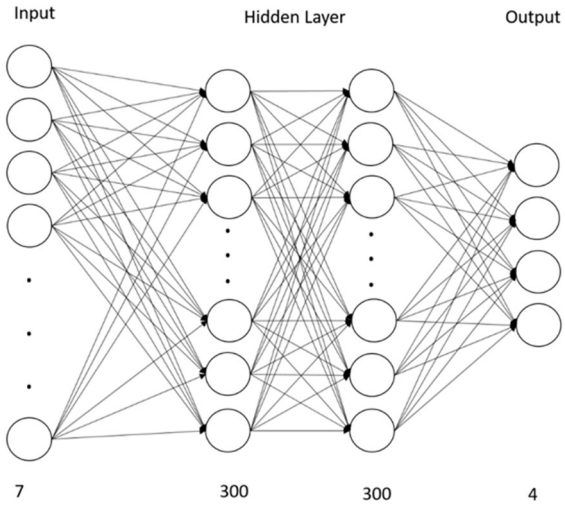

Fig. 16. Graphical representation of the training phase.

4.10. Conceptual Mining Operation Layout

The optimal pit design structure and mine layout were designed using GEOVIA GEMS [37], and the optimization was performed in Whittle [36]. The pit shell design has a total tonnage of 3,000 Mt, of which 1,566 Mt is ore with grades of 51% Al_2O_3 and 5% SiO_2.

Table presents the material that GEOVIA Whittle [36] used to generate, the designed pit boundary, and pushbacks.

Table 3. Summary of material tonnages [20]

Description	Total tonnage (Mt)	Ore tonnage (Mt)
Whittle optimum pit shell	2763.0	1610.0
Designed pit shell	3003.0	1566.0

Pushback 1	822.0	402.0
Pushback 2	1260.0	587.0
Pushback 3	912.0	544.0

The pit was divided into three phases, allowing for more individual control of the construction phases to increase cash flow and obtain higher grade zones. The pushback design was the same for all three zones, and each had a separate access ramp to make sequential exploitation easier. Fig. 17 illustrates the pushback design of the pit. Each pushback can be fully mined before moving on to the next thanks to phased mining. The area might be used as a waste dump while mining moves on to the next phase after the first phase is finished.

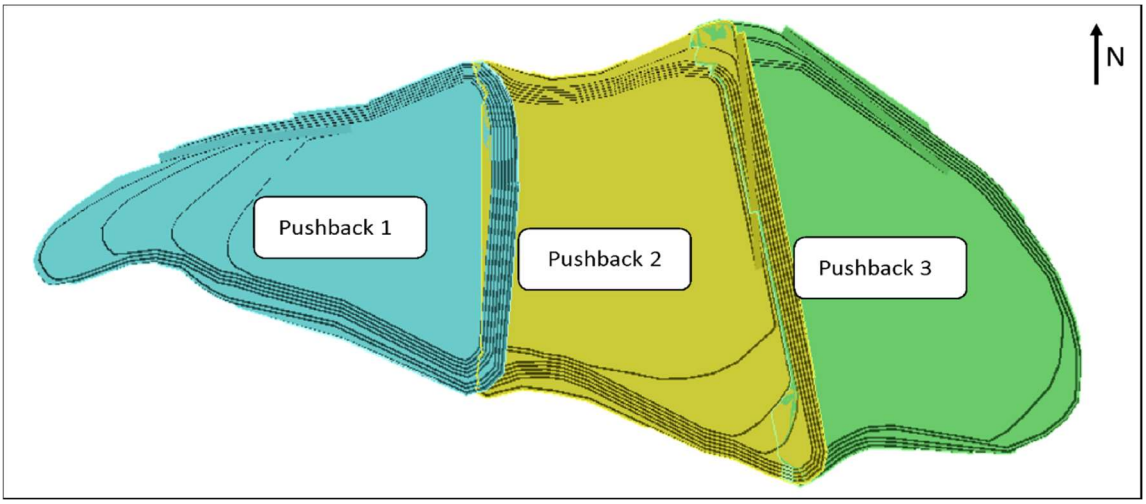

Fig. 17. Pushback designs [20]

Economic parameters considered for GEOVIA Whittle [36] to generate the final pit and pushbacks are shown in Table .

Table 4. Economic parameters [29].

Description	Value (units)
Reference mining cost	$3.16 /tonne
Reference processing cost	$9.6 /tonne
Reference stockpile cost	$0.5 /tonne
Selling price	$0.76 /%mass
Discount rate	10 %

The mine design has been designed for truck mining. The truck haulage system in the area of study has two loading points and a crushing site. The layout includes switchbacks, a road, reclaimer, a process plant, tailings, and waste dump. These settings will be used throughout the basic simulation designs. The conceptual layout is shown in Fig. 18.

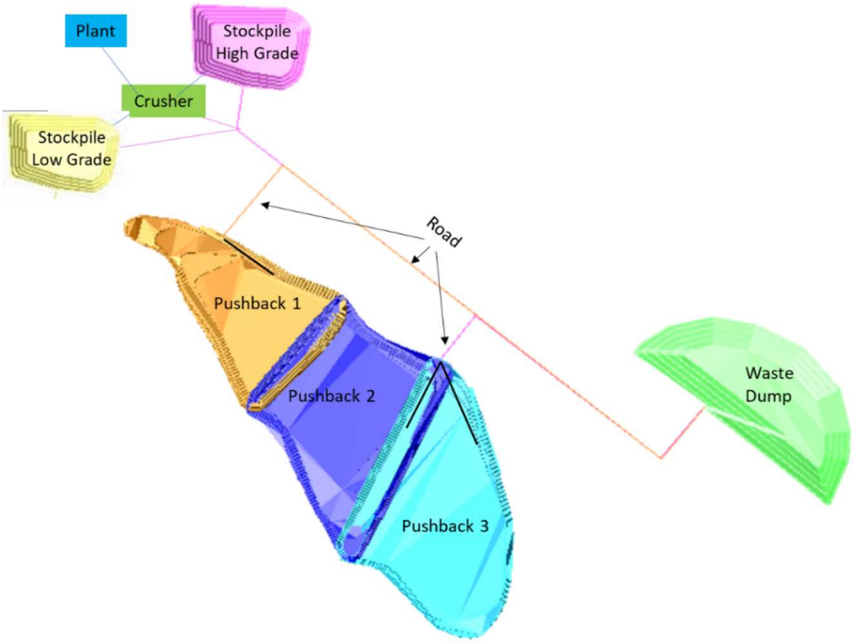

Fig. 18. The conceptual design of the operating system [20].

A strategic plan was generated using GEOVIA Whittle [36], displaying the mining, processing, and grade capacities. The mining capacity and processing capacity schedules are presented in Fig. 19 and Fig. 20 respectively.

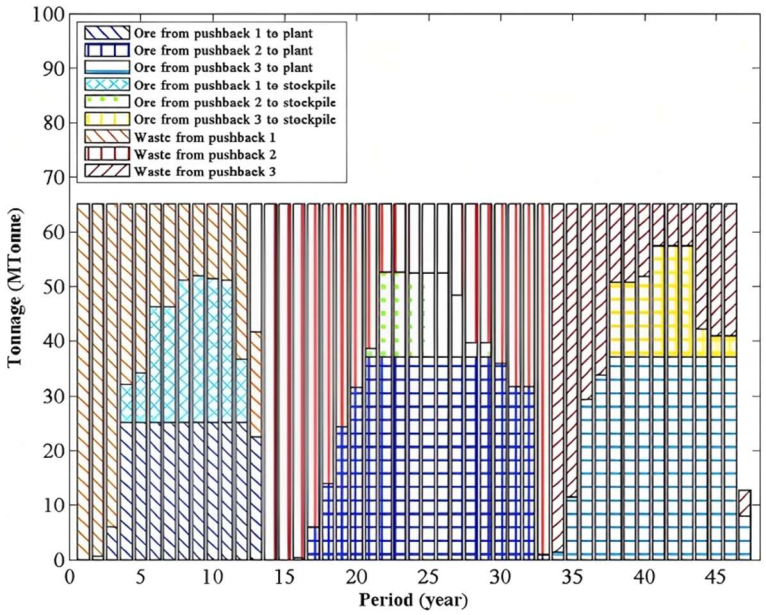

Fig. 19. Mining activity in each pushback [20].

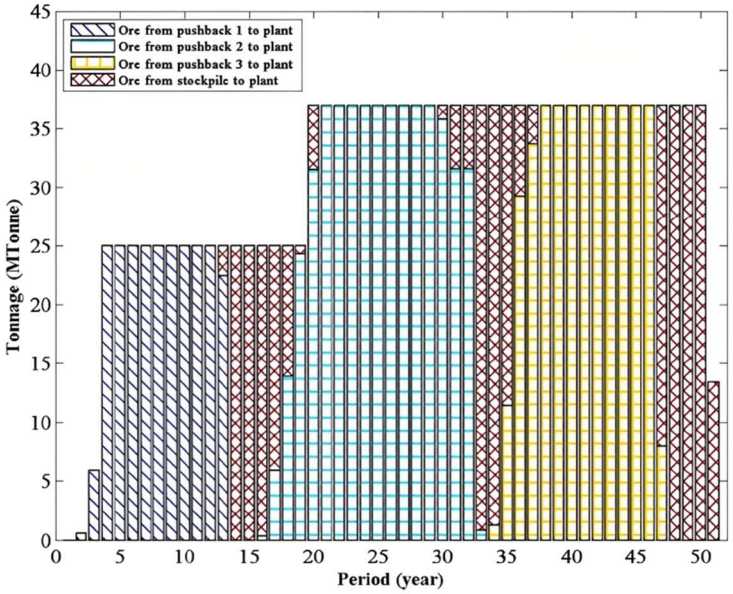

Fig. 20. Processing plant material tonnage schedule [20].

These are the parameters that GEOVIA Whittle [36] used to optimize my model.

- Life of mine is 47 years. The mining rate is 65 Mtpa.
- The processing rate from years 1 to 19 is 25 Mtpa, and 37 Mtpa from year 20 onwards.
- Processing finished in year 51.
- A stockpile is necessary to meet processing requirements.

5. Experimental Design

The design of experiments is a process with the objective of carrying out a series of tests in which deliberate changes are implemented to find out if the model behaves as expected. The testing model is like a prototype of a product, which must be tested in different ways [23]. The importance of knowing the results for each test gives us insight into the model's performance [24]. After the model is completed, verification and validation are important. Verification is the process of making sure that the model behaves as desired, and validation is the process of proving the model performs as the real system does [23].

The verification of the model was performed with a single entity following the mining block flow in the model, performing as expected. Validation was performed by changing the mining and processing ratios in the model. These ratios were decreased and increased to measure their impact on the mine life. The main effects and possible interactions between the input factors and the results were measured, concluding that the outcomes were as expected.

5.1. Cases to evaluate

There are four consecutive experiments that showcase a result in each phase. These sequential parts lay the groundwork for what comes next. The repetition creates a coherent and thorough investigation without leaving any gaps or flaws. Fig. 21 presents a schematic overview of the case study.

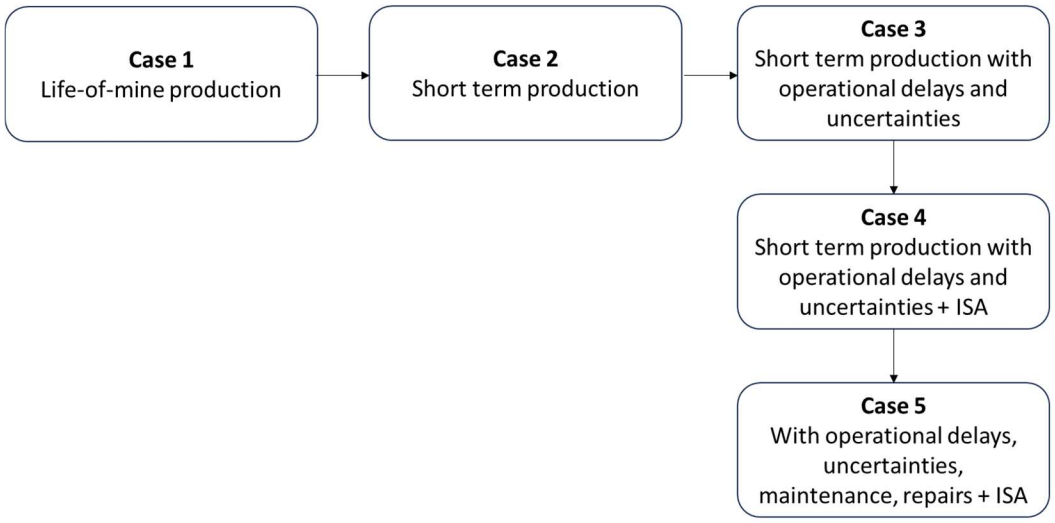

Fig. 21. Schematic overview of case study.

Only Pushback 1 data was considered in every case. The simulation was set to hours, and the following parameters were used in the application scenarios:

5.2. Case 1 – Life-of-Mine Production

The base case refers to this scenario. The intention is to replicate mining and processing operations using Anylogic software [2]. All scenarios presuppose that the mine is open every day of the year, 24 hours a day. This example replicates the realistic system, which is the GEOVIA Whittle's strategic production schedule[36].

The following configuration was set-up in the simulation model:

- Recreate mining and processing activities.
- The mining ratio is 7,420 t/h.
- The ratio for the processing plant was 2,850 t/h.

5.3. Case 2 – Short-Term Production

A short-term production schedule with weekly tonnage and grade requirements was simulated to focus on Pushback 1 Period 8. The following characteristics apply to this instance:

- There were 27 mining zones in Period 8, with 5 upper mining zones on Bench 1482 and 22 lower mining zones on Bench 1457.
- Three shovels, each with a bucket capacity of 34 m^3, were used for this experiment.
- 18 trucks, each capable of hauling 200 tons of waste and ore, were used.
- Two stockpiles (stockpiles of high and low grade)
- In this instance, no operational delays were considered.

Case 3 – Short-Term Production Schedule Simulation with Operational Delays and Uncertainties

The operating delays are added as deterministic parameters in this scenario. The following characteristics apply to this situation:

- Similar mining rate, haulage, and equipment fleet to Case 2.

- Spotting time has a normal distribution with a mean of 46 seconds and a standard deviation of 2.3 seconds.
- Mining rate per shovel is a triangular distribution with minimum, peak and maximum values of 2000 t/h 2474 t/h, 3000 t/h, respectively. This gives a total peak mining rate of 7420 t/h for 3 operating shovels.
- Processing plant feed rate is a normal distribution with a mean and standard deviation of 2853 t/h, 285 t/h, respectively.
- The haulage model uses a 3D design of the haulage profile imported from Geovia Gems. A normal distribution with a mean and standard deviation of 37, 3 respectively, is used for the truck speed.

Case 4 – Short-Term Production Schedule Simulation with Operational Delays, Uncertainties, and ISA Integration

Case 4 discusses the experimental behavior of the ISA from Case 3 when uncertainties are introduced in the mining system. The objective of this case is to test the performance of the ISA when launched in an environment with stochastic parameters. The operational parameters were similar to the mining and processing rates and operational uncertainties from Case 2.

Case 5 – Short-term Production Schedule Simulation with Operational Delays, Uncertainties, Maintenance, Repairs, and ISA Integration

Case 5 is a combination of Case 4 with the addition of equipment failure maintenance and repairs and ISA integration. In this experiment, we included all operational parameters to test the performance of the ISA. Mining and processing rates were similar to Case 2. Table illustrates the operational and uncertainty parameters for the simulation. The normal parameter takes the input value as the mean and the square root of the mean value as its standard deviation.

Table 5. Parameters for equipment maintenance schedule, failure, and repair.

	Failure	Maintenance	Repair
Shovels	Uniform distribution (1- 180hrs failure event)	250 hrs. maintenance schedule	Normal distribution (2hrs repair time)
Plant	Uniform distribution (1 - 5 weeks)	3 months maintenance schedule time Normal distribution (10hrs – 12hrs maintenance time).	Normal distribution (100 minutes repair time)
Trucks	Uniform distribution (1 -38hrs failure event)	168 maintenance schedule events 2 hrs. maintenance time	Normal distribution (1 - 2hrs repair time)

6. Results

6.1. Case 1 – Life-of-mine production

This case was created to replicate the long-term production schedule for Pushback 1 from Whittle Optimization.

6.2. Case 2 – Short term production

The evaluation for this scenario was done on a short-term basis for Pushback 1 and Period 8. According to the mining plan, 65 Mt needed to be mined. Fig. 22 shows the mining production schedule. The simulation finished in Week 53. The mining production is steady throughout all periods; however, the ore drops in Weeks 9 and 10. Fig. 23 shows a schedule for weekly mining production based on shovel activity. Fig. 24 illustrates the plan view of mining according to zones and shovel allocation. Mining commences in Zone 1 and ends in Zone 27 in a sequential form. Fig. 25 shows the weekly schedule for plant feeding from stockpiles and pit mining. The plant processed

51 Mt of ore in total. Fig. 26 is the stockpile inventory, which demonstrates how materials accumulate over time in the high- and low-grade stockpiles. In general, material is reclaimed to the plant after both stockpiles reach a maximum of 26 Mt in Week 52.

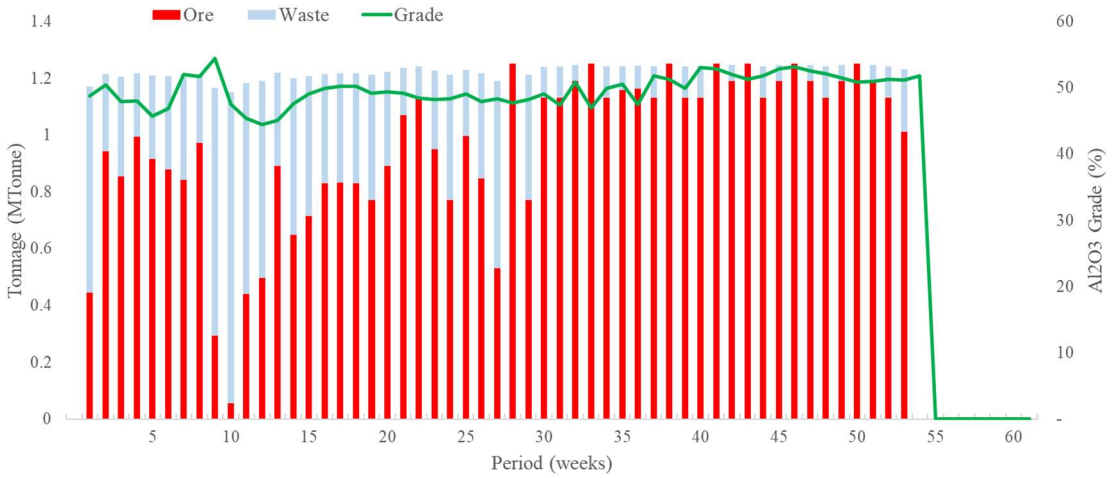

Fig. 22. Weekly mining production schedule showing material types.

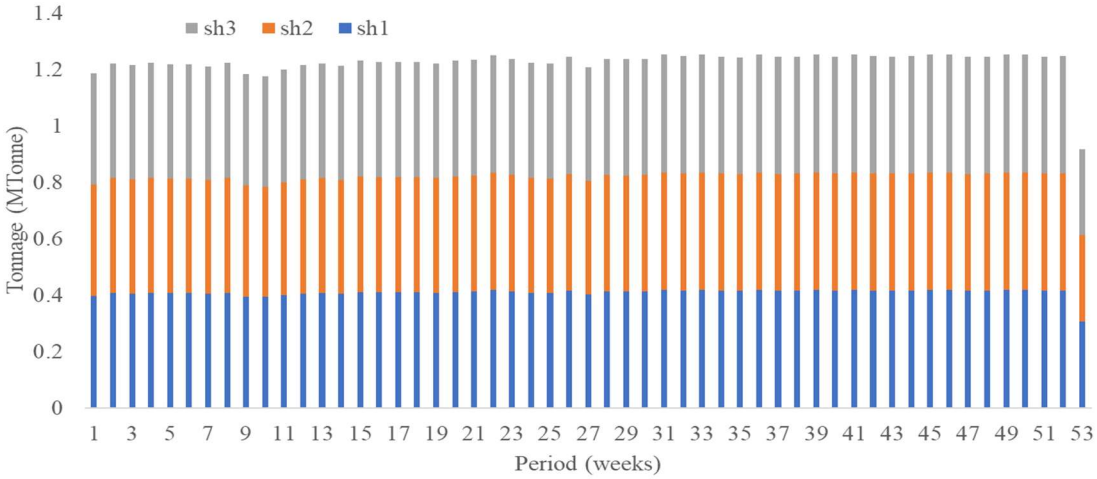

Fig. 23. Weekly mining distribution schedule showing zones.

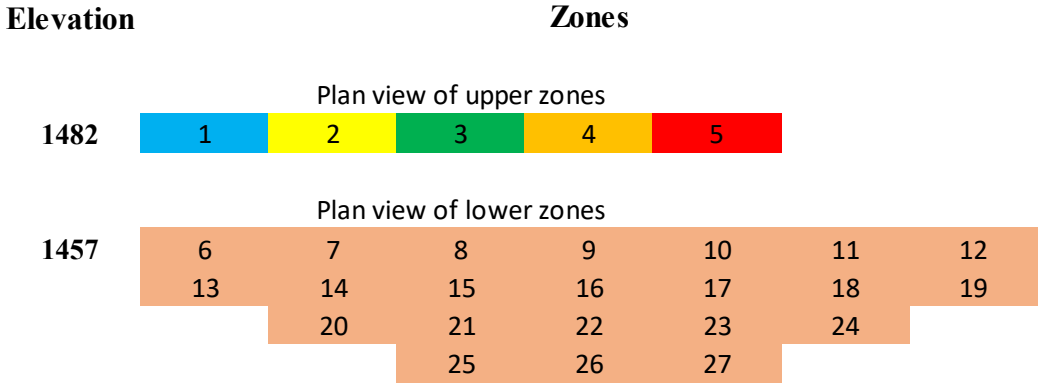

Fig. 24. Plan view of mining zones for shovel allocation.

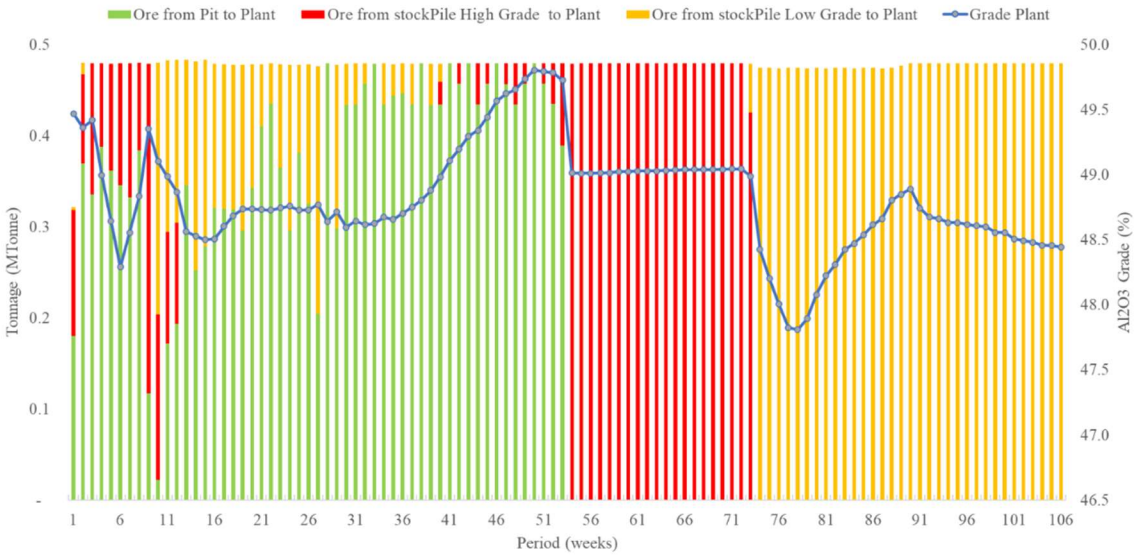

Fig. 25. Weekly plant feed schedule showing material sources.

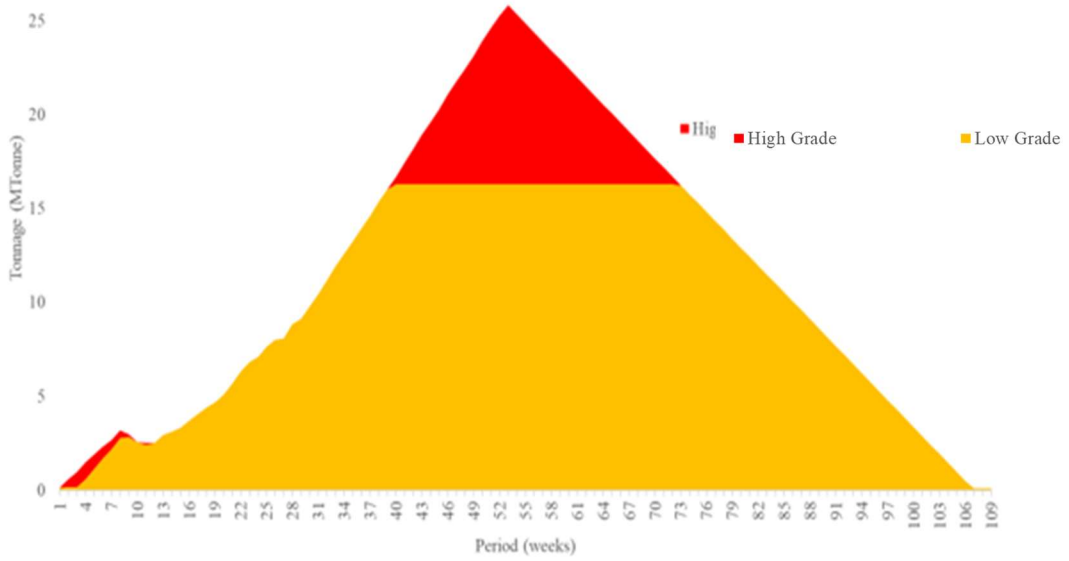

Fig. 26. Stock inventory showing high- and low-grade material.

6.3. Case 3 – Short-term Production Schedule Simulation with Operational Delays and Uncertainties

Fig. 27 illustrates the weekly mining production schedule by material types, and Fig. 28 shows the weekly mining production schedule by shovel activity. Fig. 29 illustrates the weekly mining production schedule by mining zones following a predetermined extraction sequence like Case 2. The pit mining operation for ore and waste lasted for 54 weeks. In Week 10, only waste rocks were extracted from the pit, which was supplemented by material from the stockpiles. Fig. 30 illustrates the weekly plant feed schedule from pit mining and stockpile reclamation lasting for 106 weeks with a steady weekly processing rate of 480,000 tons. Fig. 31 shows how the stockpile inventory is accumulated in the high-grade and low-grade stockpiles, reaching a peak of 26 Mt in Week 52, followed by reclamation to the plant after pit mining ends.

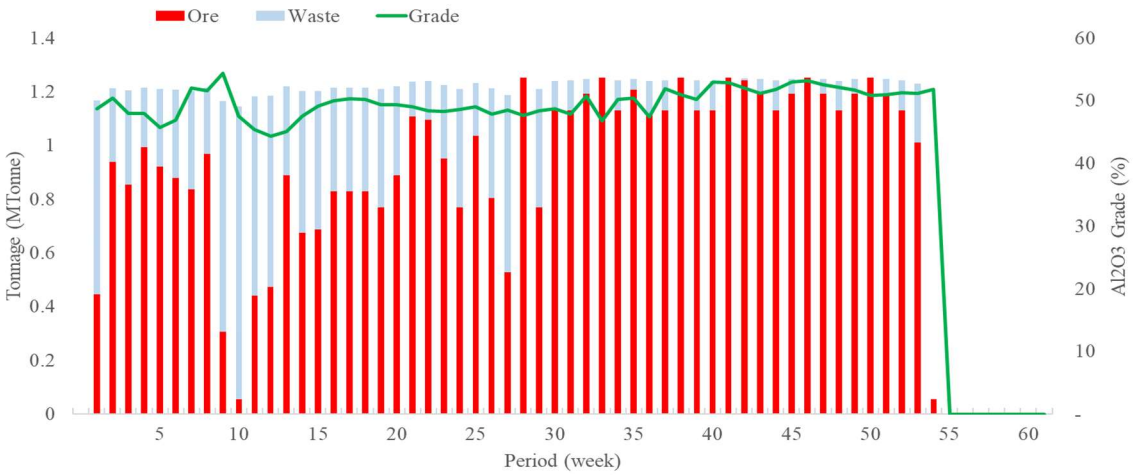

Fig. 27. Weekly mining production schedule showing material types.

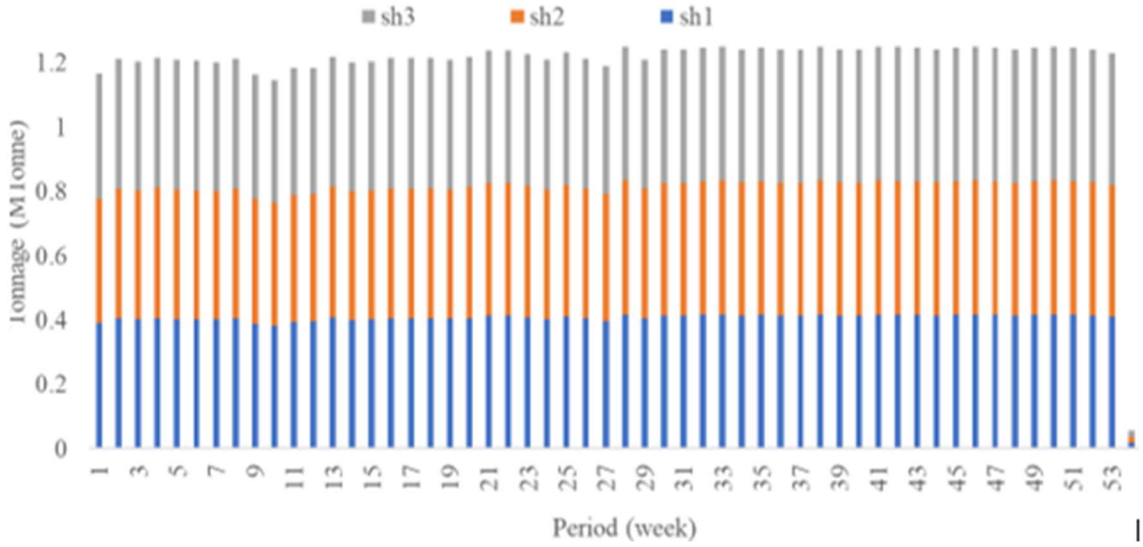

Fig. 28. Weekly mining production schedule showing shovel tonnes.

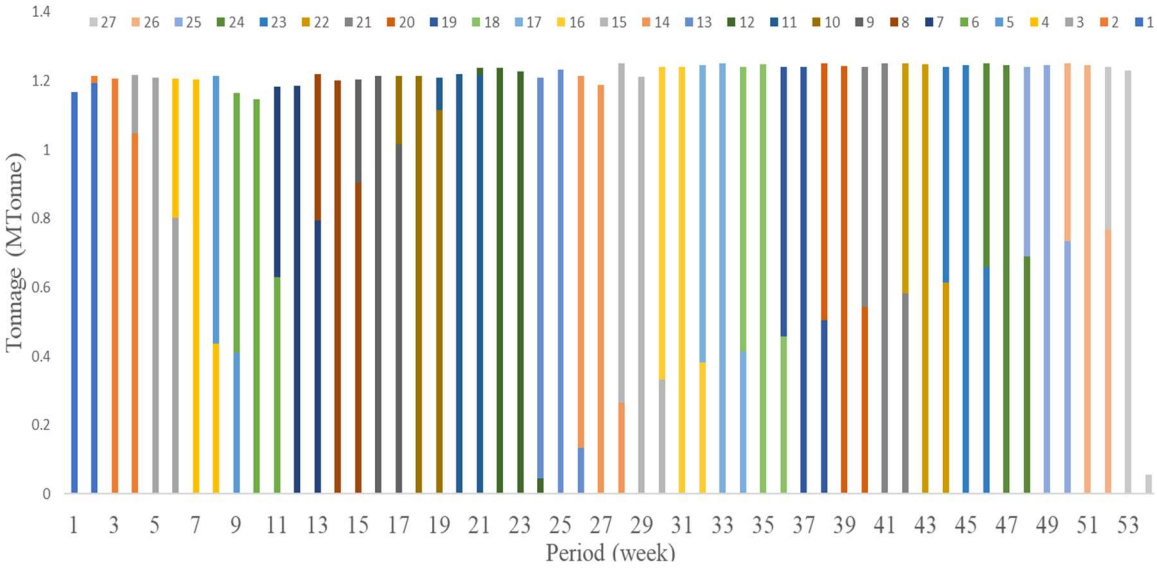

Fig. 29. Weekly mining production schedule showing mining zones.

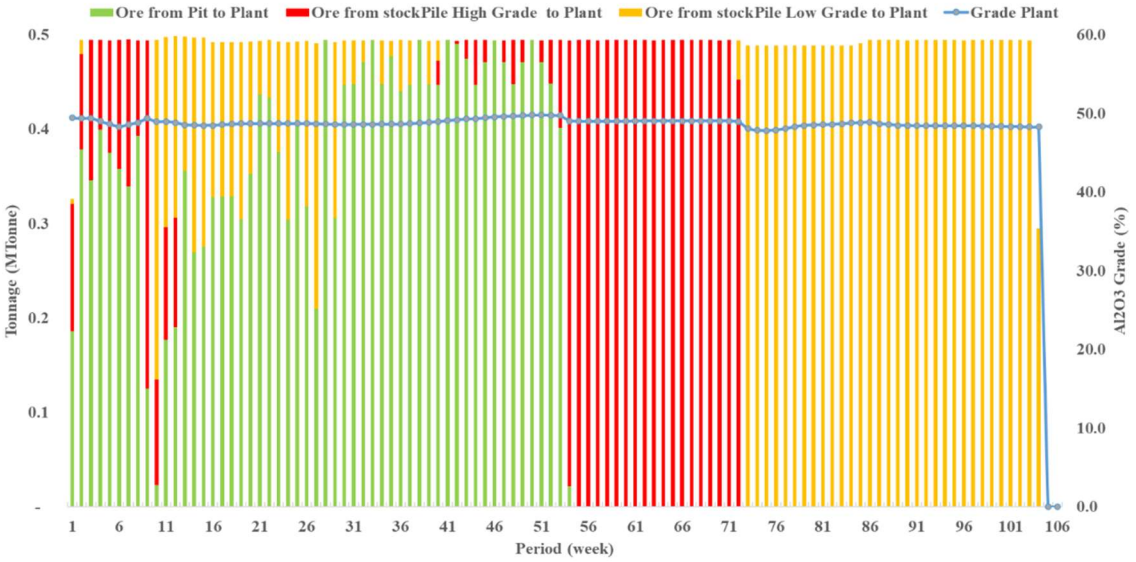

Fig. 30. Weekly plant feed schedule showing material sources.

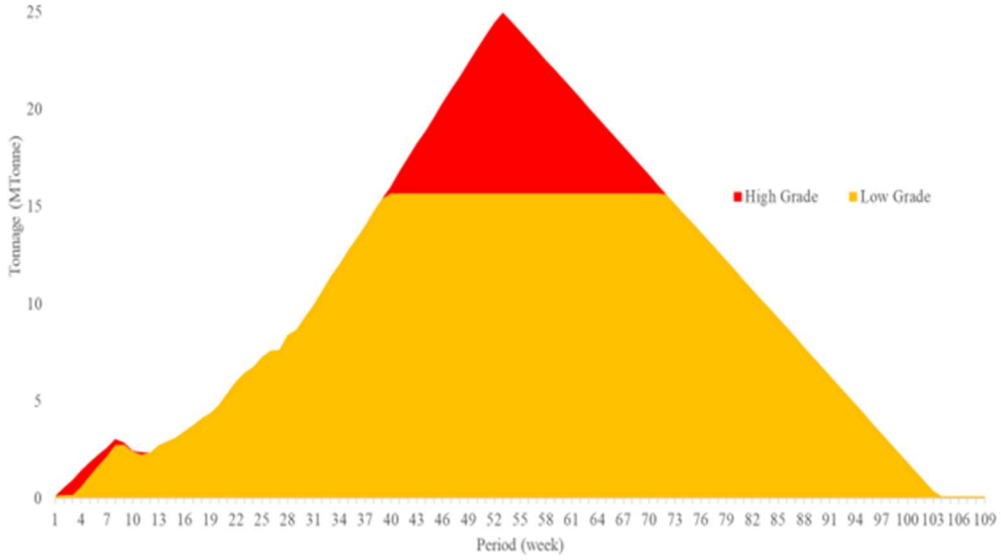

Fig. 31. Weekly stockpile inventory showing high and low grade material.

6.4. Case 4 – Short-term Production Schedule Simulation with Operational Delays, Uncertainties, and ISA Integration

The weekly mining production schedule for material types is shown in Fig. 32. Pit mining took 55 weeks, with only two shovels working from Week 52 to Week 55. This was due to the ISA's restriction on one shovel per zone at any time. The haulage distance from the pit to the waste dump is longer, resulting in reduced shovel productivity. Fig. 33 is the weekly mining production schedule showing mining zones. It shows when each mining zone is extracted based on the priorities and choice of actions of the ISA. The weekly plant feed schedule in Fig. 34 shows material from the pit and stockpiles. Fig. 35 shows the stockpile inventory and how material accumulates in both the high- and low-grade stockpiles reaching a peak of 26 Mt in Week 52 before reclamation to the plant. The ISA controls the reclamation of materials by blending high- and low-grade materials for a stable plant feed grade.

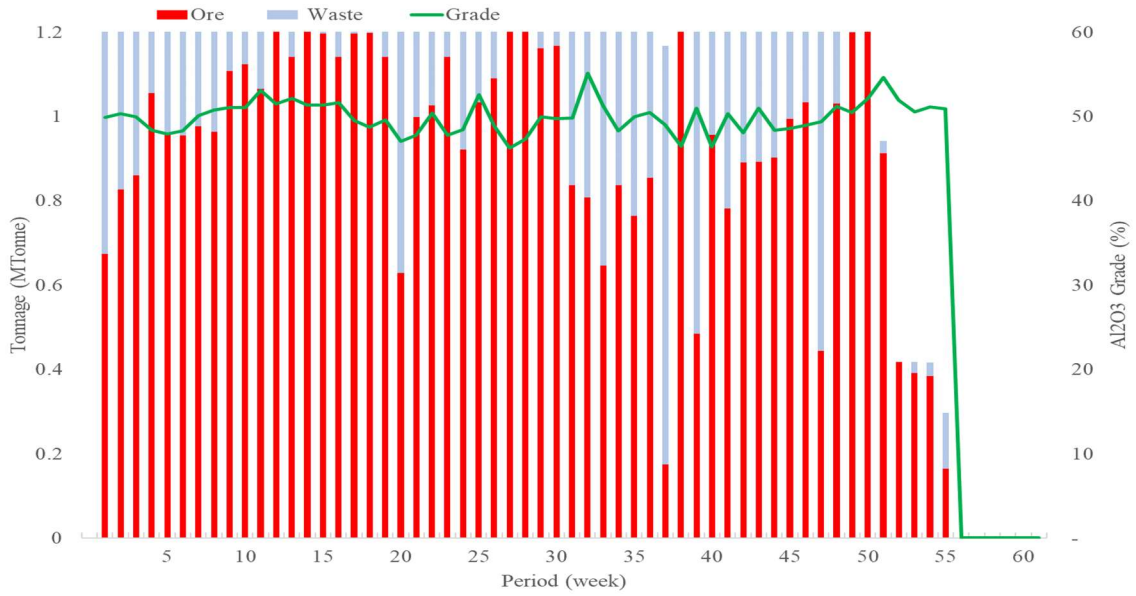

Fig. 32. Weekly mining production schedule showing material types.

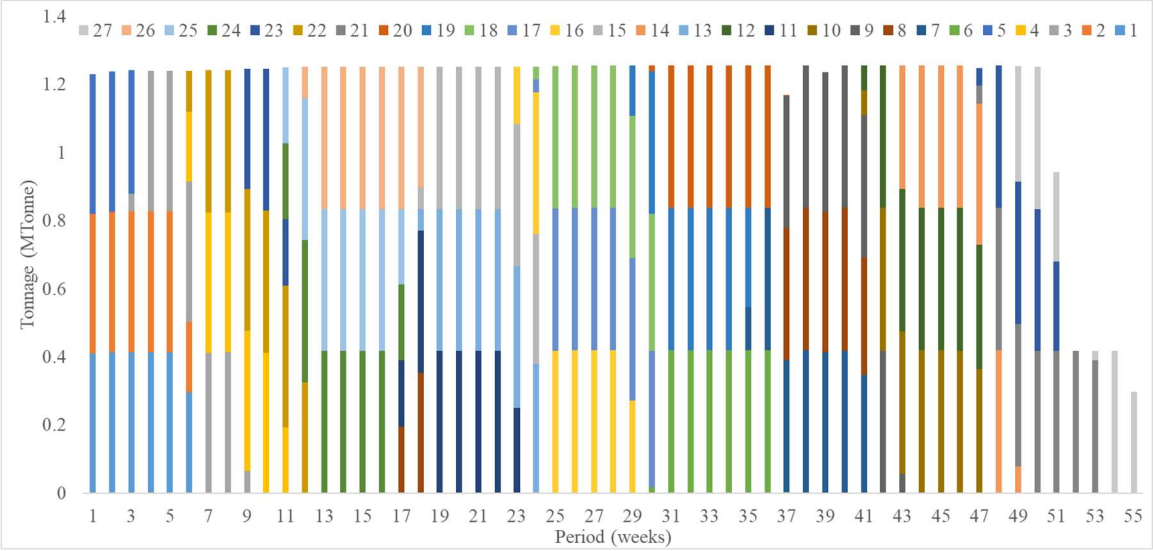

Fig. 33. Weekly mining production schedule showing mining zones.

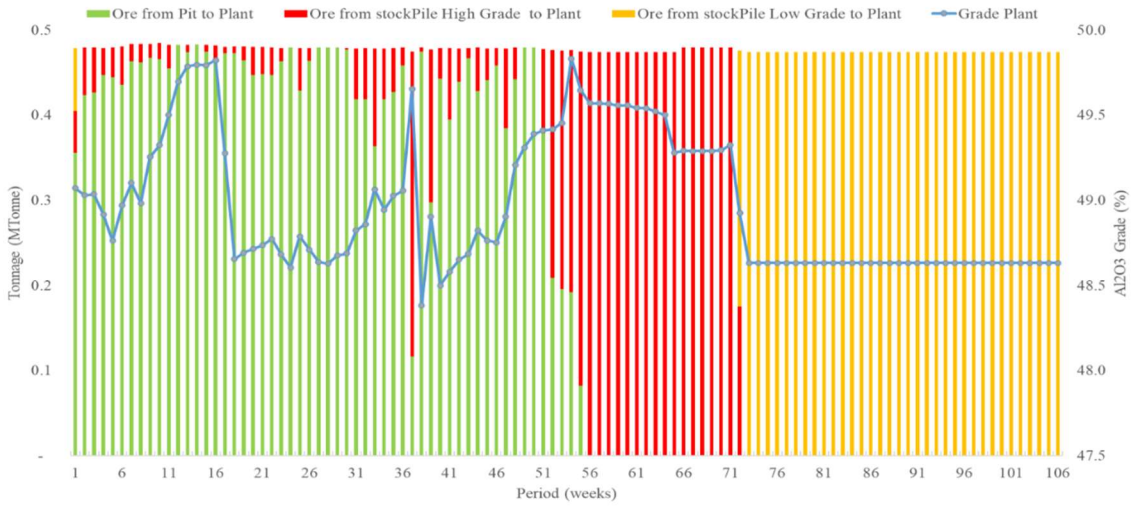

Fig. 34. Weekly plant feed schedule showing material sources.

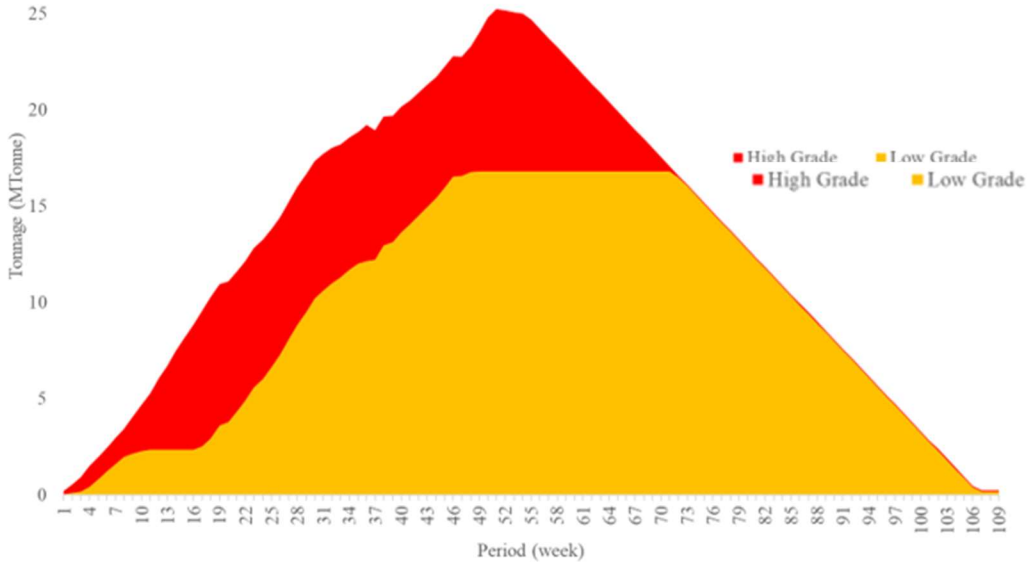

Fig. 35. Weekly stockpile inventory showing high- and low-grade material.

6.5. Case 5 – Short-term Production Schedule Simulation with Operational Delays, Uncertainties, Maintenance, Repairs, and ISA Integration

Fig. 36 is the weekly mining production schedule showing ore and waste material mined over the period. Pit mining lasted for 56 weeks, and only one shovel worked from Week 52 to Week 56. The ISA is restricted to allowing only one shovel to work in a zone at any point in time. Fig. 37 is the weekly mining production schedule showing mining zones. It shows when each mining zone is extracted based on the priorities and choice of actions in the ISA. Fig. 38 illustrates the weekly plant feed schedule, showing material from pit mining and stockpiles. The consistent shortfalls over the period were due to the failure and maintenance of the plant. Fig. 39 shows the stockpile inventory. The stockpile accepts material when the crusher is busy. It stores high- and low-grade materials, which are then reclaimed by the plant when pit mining ends.

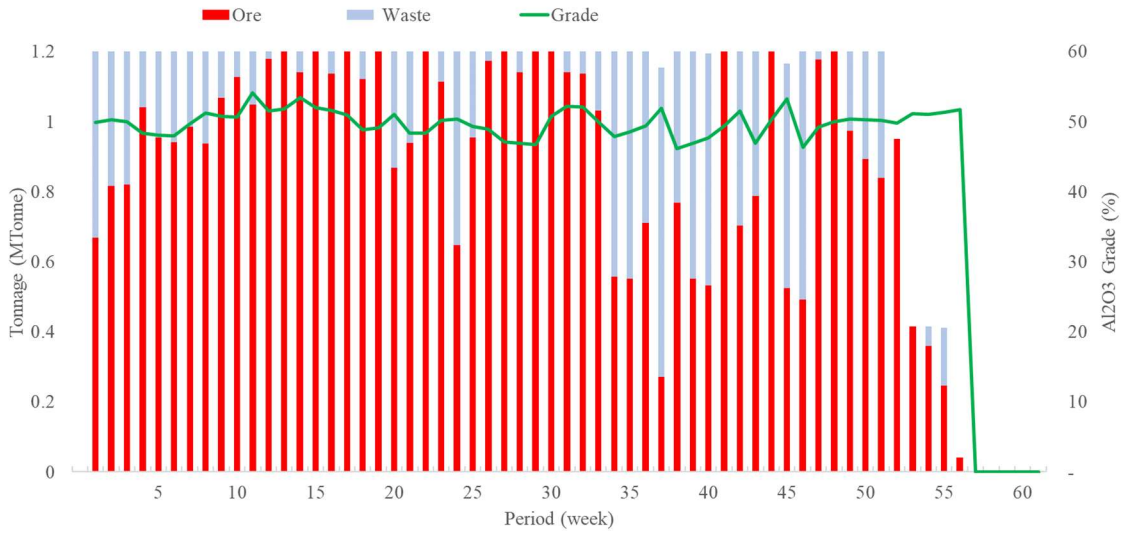

Fig. 36. Weekly mining production schedule showing material types.

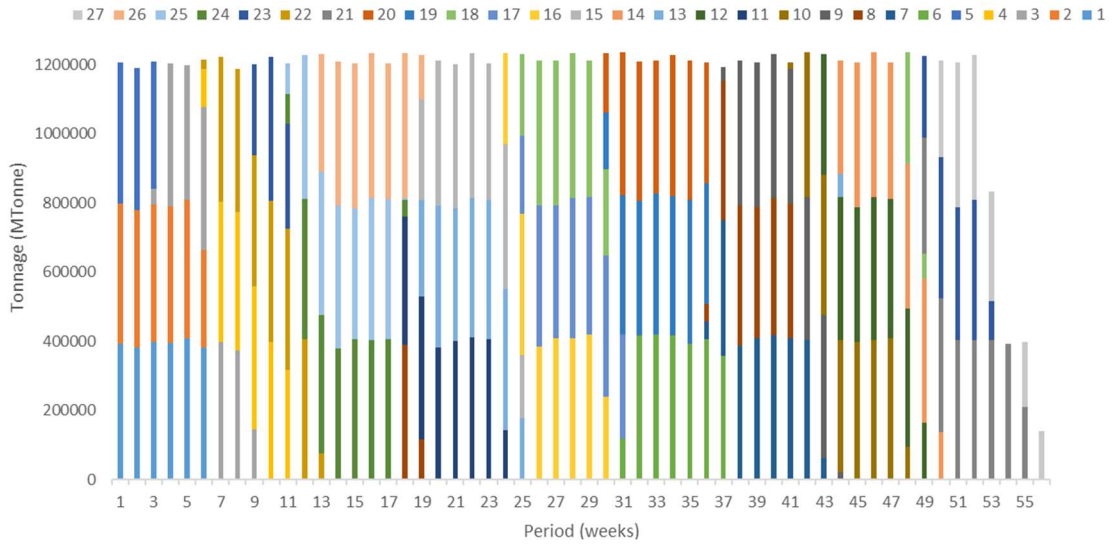

Fig. 37. Weekly mining production schedule showing mining zones.

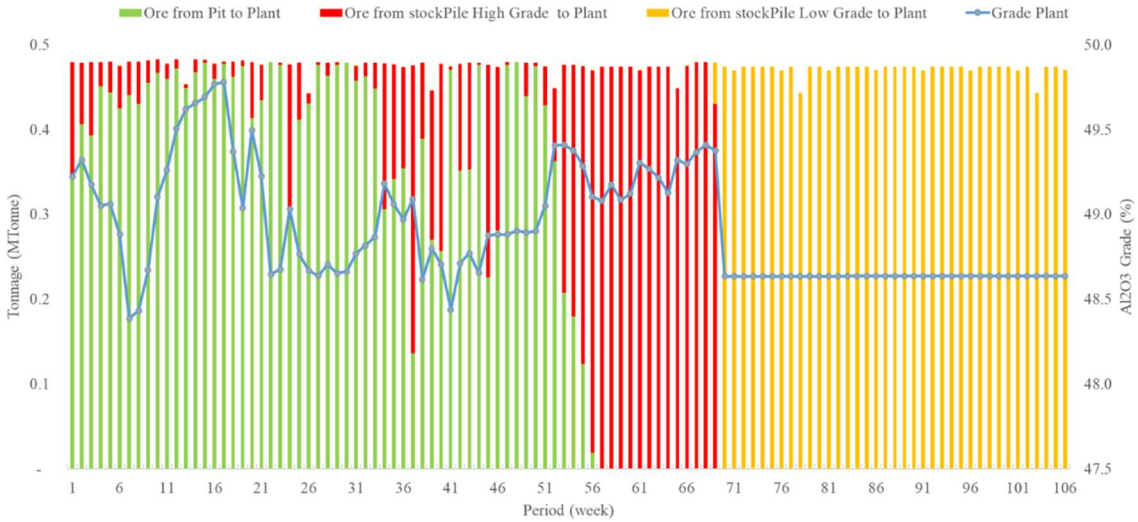

Fig. 38. Weekly plant feed schedule showing material sources.

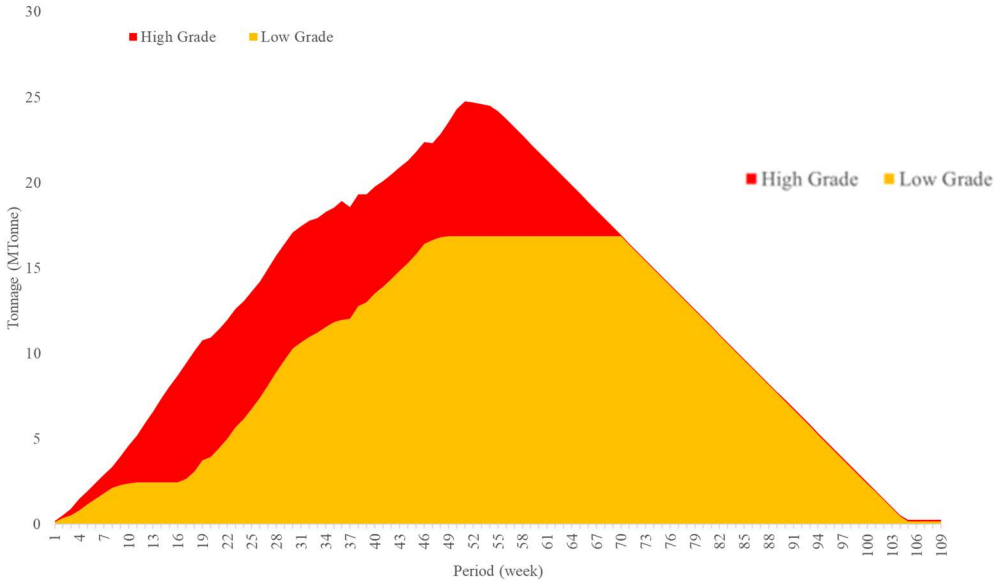

Fig. 39. Weekly stockpile inventory showing high- and low-grade material.

7. Comparison experimental results and discussion

Fig. 40 represents the weekly mining production, and Fig. 41 shows the grade of ore tonnage mined according to the experimental cases. The ISA did well to improve mining production in the early weeks. Fig. 42 illustrates the weekly ore tonnage processed. There are observed shortfalls in plant processing for Case 5. These shortfalls are attributed to plant failure and a poor maintenance schedule. Fig. 42 shows the ore tonnage processed which achieved a steady 480,000t/week for Case 3. The shortfalls in Case 5 are attributed to the equipment failure and maintenance for plant. In Fig. 43, an increase in grade occurred in the early weeks for Cases 4 and 5 due to the ISA performing grade blending from the stockpile. Fig. 44 shows the metal content of the ore processed is higher than in Case 4 despite the lower ore tonnage. Comparing Case 5 to the other cases after period 50, Case 5 has lower stockpile levels, as shown in Fig. 45.

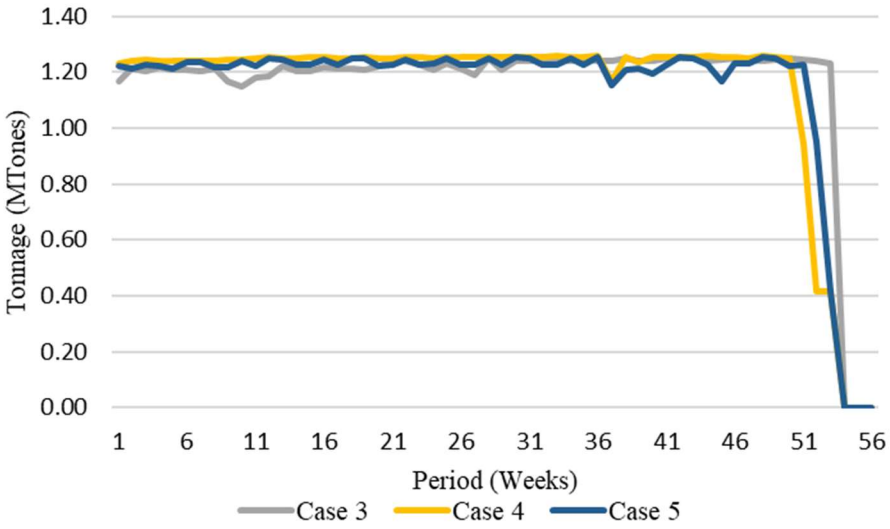

Fig. 40. Comparisons of experimental cases for mining production.

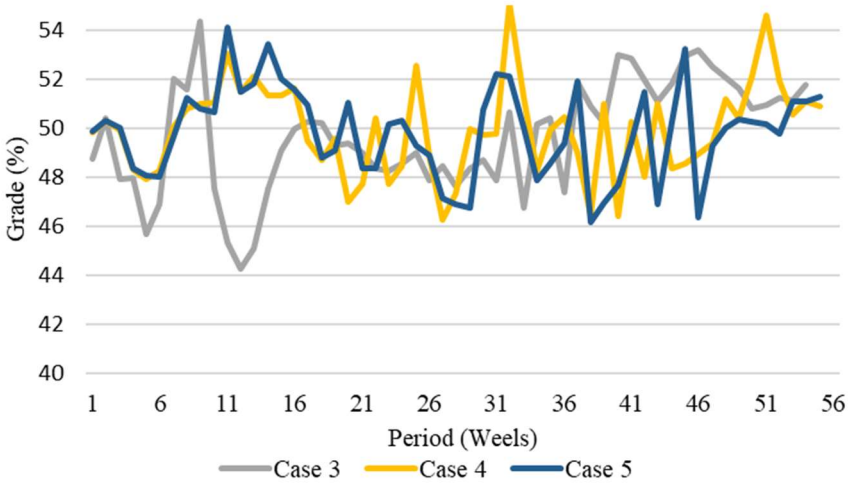

Fig. 41. Comparisons of experimental cases for ore mining grade.

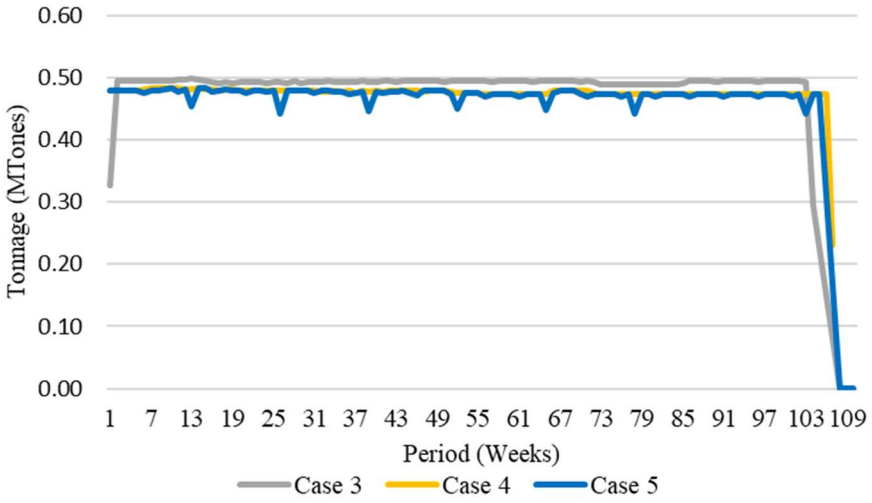

Fig. 42. Comparisons of experimental cases for ore tonnage processed.

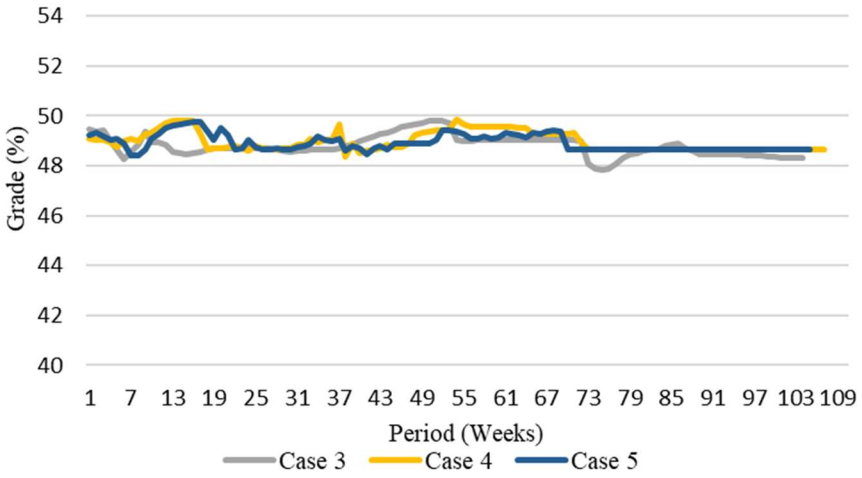

Fig. 43. Comparisons of experimental cases for processing plant head grade.

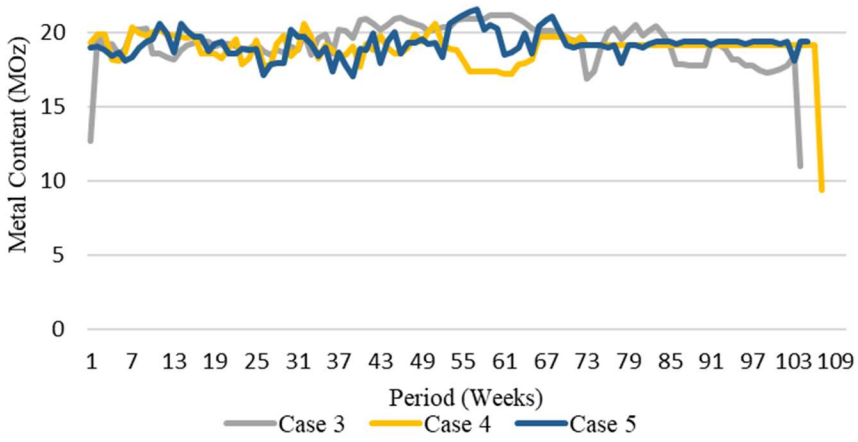

Fig. 44. Comparisons of experimental cases for metal content for ore processed.

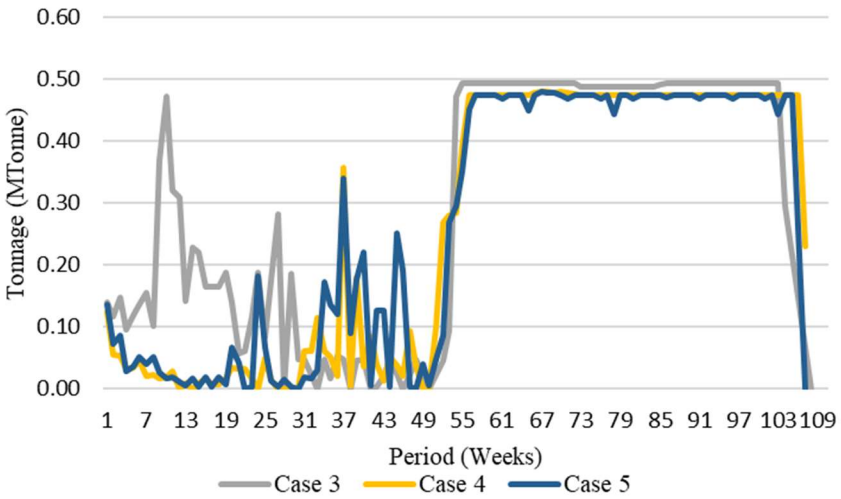

Fig. 45. Comparisons of experimental cases for stockpile material.

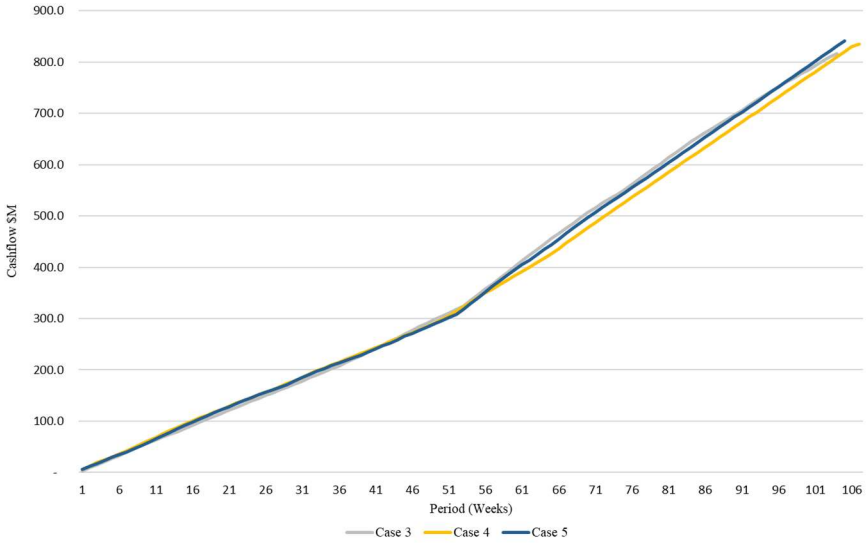

Fig. 46. Comparisons of experimental cases for cumulative cash flow.

The cumulative cash flows for various experimental cases over time are displayed in Table . For each experimental case, the cumulative cash flow rises over time, which is positive and demonstrates that income exceeds expenses. The Case 5 cash flow total is the highest at $840.9 million after 104 weeks. It appears from this that Case 5 is the most lucrative. The cumulative flows between the cases range from $815.9 million to $840.9 million, so the difference is not very significant. This suggests that the cases financial results are comparable.

Table 6. Cumulative cashflow for experimental cases.

Experimental Case	Period (Weeks)	Cumulative Cashflow ($M)
3	104	815.9
4	107	834.6
5	104	840.9

Fig. 47 to Fig. 50 illustrates truck and shovel utilizations for the experimental Cases 4 and 5. A steady utilization of shovels is observed in Case 4, since the shovels were always considered available, and mining continued until all blocks were exhausted. The utilization data for trucks and shovels ceases recording once mining is completed. Shovels 2 and 3 start to decline from Week 50 as all mining

zones have been exhausted except for the last zone which has been allocated to Shovel 1. A reduction in Case 5 compared to Case 4, mainly due to shovel availability, maintenance, and repairs.

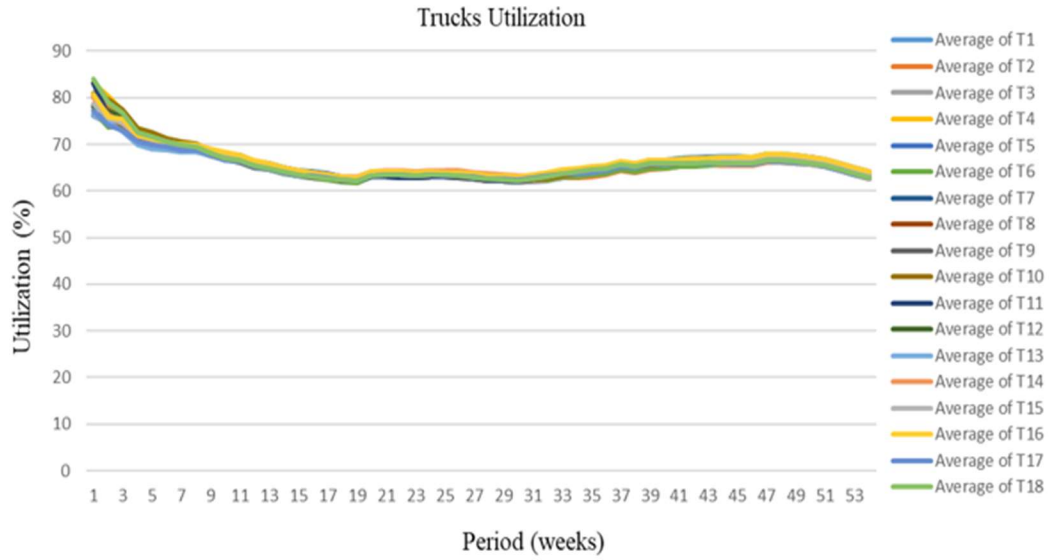

Fig. 47. Case 4 – Truck utilization.

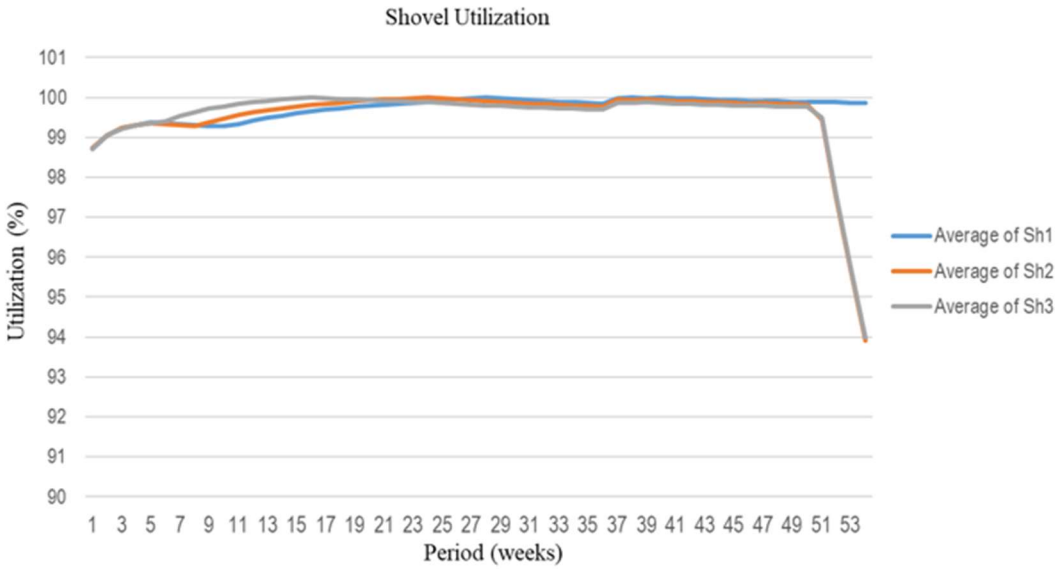

Fig. 48. Case 4 – Shovel utilization.

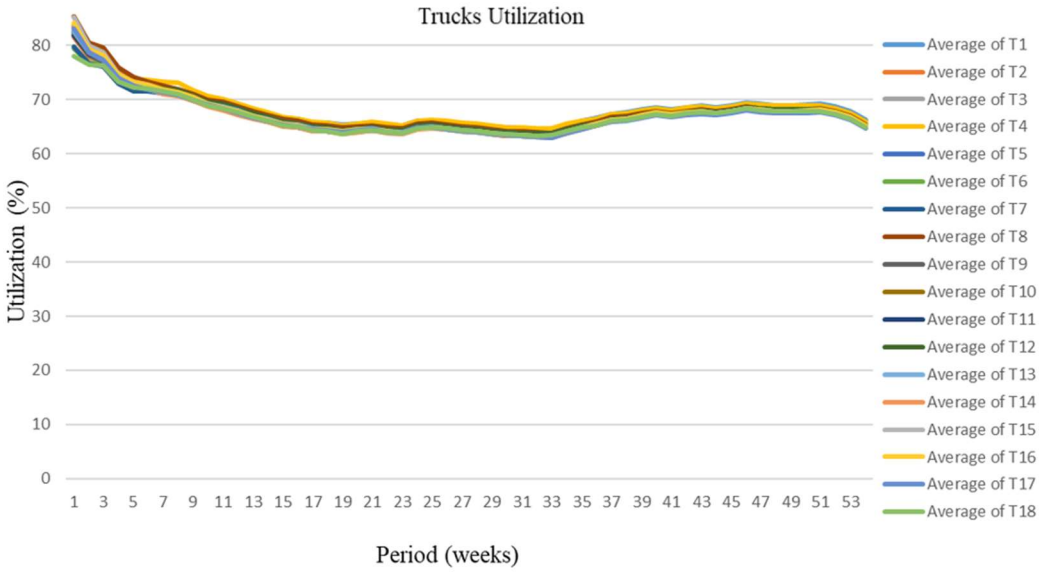

Fig. 49. Case 5 – Truck utilization.

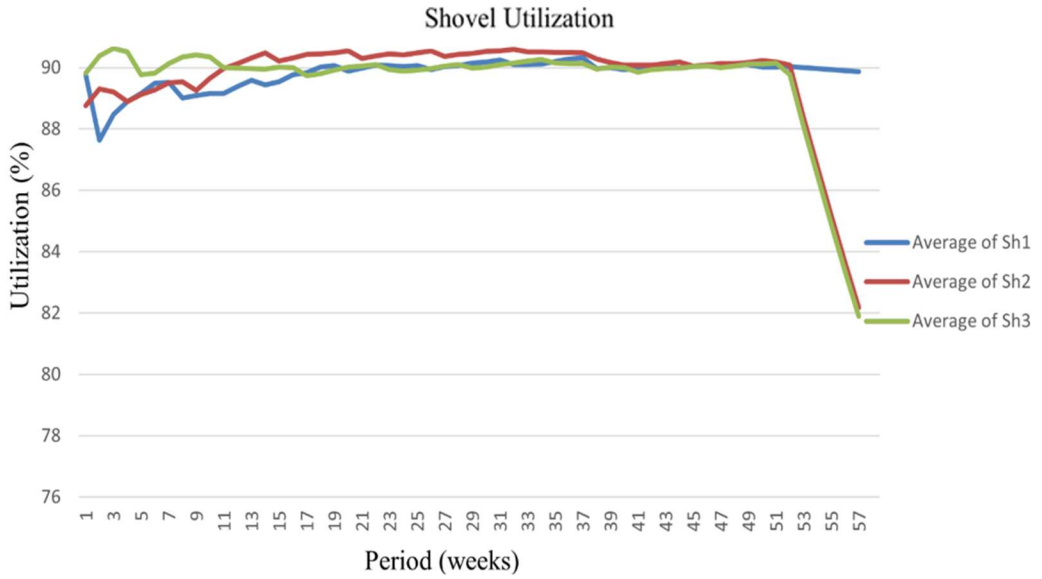

Fig. 50. Case 5 – Shovel utilization.

The outcomes showed how the Intelligent Supervisory Agent (ISA) could maintain and improve productivity even in the face of growing uncertainties and disruptions across the cases. Case 3 added more realism by including delays and uncertainties related to mining and processing parameters, but the activities were still completed according to the planned schedules. Case 4 demonstrated the ISA's abilities to monitor grade control, smooth mining production, and restore stockpiles in the face of variations. The ISA accomplished this by allocating resources dynamically by prioritizing zones based on ore tonnage and grade. By modifying shovel schedules and utilizing stockpiles to meet targets, Case 5 demonstrated the ISA's capacity to adapt to equipment failures and maintenance activities, resulting in the highest cashflow. When compared to Case 4, shovel utilization dropped in Case 5, but the overall output was still optimized. The development of the cases showed that relying solely on fixed schedules leaves no room for flexibility in the face of uncertainties, whereas intelligent systems like the ISA enable real-time adaptive decision-making to maintain productivity and increase cash flow. The importance of AI-driven technologies for efficiency in difficult mining environments is highlighted by this.

8. Conclusions

The simulation environment chosen for modelling Smart Mining Automation (SMA) involves a combination of discrete event simulation (DES) and an agent-based model (ABM). To train the intelligent supervisory agent (ISA) mentioned in this study, the selected reinforcement learning algorithm is the Deep Q network. The proposed SMA was implemented for a short-term mine plan and a bauxite mining operation, which included operational decisions taken on a weekly basis. Based on the results obtained, the following conclusions were drawn:

1. The proposed SMA can simulate a real-life mining operation, and with the integrated ISA, operational decisions such as block-sequencing and truck-shovel dispatching are made. The ISA performs block sequencing by prioritizing mining zones with more ore tons and high grades and relocating shovels to available mining zones when ore production is below the weekly KPI targets. Truck dispatching by the ISA is done based on the availability of working shovels.
2. The use of an ABM for modelling the mining resources (shovels, trucks, and plants) closes the gap of many input states fed to the neural network of the ISA. ABMs interact with each other as well as their environment. Since they already have individual states defining their behaviour, controlling the individual states with an integrated central neural network becomes robust.
3. The research demonstrates how operational decisions can be taken by a trained RL agent to maximize productivity without the intervention of human. Its significant contribution is the real-time decision-making capacity that it acquires as the mining simulation progresses through time.
4. The use of multiple stockpiles in the mining operation helped reduce the mining and processing times by storing high- and low-grade material when the crusher queue was larger than the set limit.
5. With the introduction of the ISA, fewer labor meaning less interaction between people and equipment, can be achieved, leading to improvements in health and safety records and financial savings.

9. References

[1] Alipour, A., Khodaiari, A. A., Jafari, A., and Tavakkoli-Moghaddam, R. (2018). Uncertain production scheduling optimization in open-pit mines and its ellipsoidal robust counterpart. *International Journal of Management Science and Engineering Management*, *13* (4), 245-253.

[2] AnyLogic (2021). Anylogic 8.7 Personal Learning Edition. Retrieved January 05, 2022, from: https://www.anylogic.com/

[3] Ares, G., Castañón Fernández, C., Álvarez, I. D., Arias, D., and Díaz, A. B. (2022). Open Pit Optimization Using the Floating Cone Method: A New Algorithm. in *Minerals*, vol. 12

[4] Askari-Nasab, H. (2006). Intelligent 3D interactive open pit mine planning and optimization. Ph.D Thesis, University of Alberta, Edmonton, Alberta, Pages 167.

[5] BANKS, J., 1939 2001. *Discrete-event system simulation*, Upper Saddle River, NJ : Prentice Hall.

[6] Bellman, R. E. and Dreyfus, S. E. (1962). *Applied dynamic programming.* Princeton University Press, Princeton, NJ, Pages 363.

[7] Bodon, P., Fricke, C., Sandeman, T., and Stanford, C. (2011). Modelling the mining supply chain from mine to port: A combined optimization and simulation approach. *Journal of Mining Science volume 47* (2), pp. 202-211.

[8] Brownlee, J. (2021). Deep Learning. Retrieved from: https://machinelearningmastery.com/

[9] Chaturvedi, M. (2021). What are dqn reinforcement learning models. Analytics India Magazine.
[10] Choudhary, A. (2019). A hands-on introduction to deep q-learning using openai gym in python. Analytics Vidhya. .
[11] DeepMind (2021). What if solving One problem could UNLOCK solutions to thousands more? Deepmind. . Retrieved from: https://deepmind.com/
[12] Gagniuc, P. (2017). *Markov Chains: From Theory to Implementation and Experimentation.* John Wiley & Sons, New Delhi, India, 1st Edición ed, Pages 256.
[13] Glockner, C. (2018). Simulators: The KEY training environment for applied deep reinforement learning. Retrieved from: https://towardsdatascience.com/simulators-the-key-training-environment-for-applied-deep-reinforcement-learning-9a54353f494f.
[14] Gomez, J.O., García, J.A.J., Arriaga, A.V.B. and Lira, A.J.R., 2018 (2018). Analysis of the impact of the increase of production volume on the material supply system and work in process through simulation scenarios. *International Journal of Supply Chain and Operations Resilience*, *3* (3), 260-273.
[15] Gou, S. (2019). DQN with Model-Based Exploration: Efficient Learning on Environments with Sparse Rewards.
[16] Grigoryev, I. (2019). AnyLogic in three days a quick course in simulation modelling, 5th ed.
[17] Hashemi, A. S. and Sattarvand, J. (2015). *Application of ARENA simulation software for evaluation of open pit mining transportation systems–a case study*. in Proceedings of Proceedings of the 12th International Symposium Continuous Surface Mining-Aachen 2014, Springer, pp. 213-224.
[18] Hebb, D. O. (2005). *The organization of behavior: A neuropsychological theory.* Psychology press.
[19] Holford, N. H., Hale, M., Ko, H., Steimer, J., Sheiner, L., Peck, C., Bonate, P., Gillespie, W., Ludden, T., and Rubin, D. (1999). Simulation in drug development: good practices. *Center for Drug Development Science (CDDS)*, *1* 23.
[20] Ben-Awuah, E. and Hosseini, N.S., 2017. An economic evaluation of a primary haulage system for a Bauxite mine: load and haul versus in-pit crushing and conveying. Min. Optim. Lab, 1, p.109.
[21] Hustrulid, W. A., Kuchta, M., and Martin, R. K. (2013). *Open pit mine planning and design, two volume set & CD-ROM pack.* CRC Press.
[22] Jovanoski, B., Polenakovik, R., Gecevska, V. and Minovski, R., (2014). *Applying a suitable simulation approach for processes on different management levels*. in Proceedings of XVI INTERNATIONAL SCIENTIFIC CONFERENCE ON INDUSTRIAL SYSTEMS, University of Novi Sad - Faculty of Technical Sciences Department of Industrial Engineering and Management, Novi Sad, Serbia., pp. 327-332.
[23] Kelton, W. D., Sadowski, R. P., & Zupick, N. B. (2015). *Simulation with arena.* McGraw-Hill Education, New York, N.Y., Sixth edition ed, Pages 635.
[24] Lazic, L. and Velasevic, D. (2004). Applying simulation and design of experiments to the embedded software testing process. *Software testing, verification & reliability*, vol. *14* (4), pp. 257-282.
[25] Lee, J., Won, J., and Lee, J. (2018). Crowd simulation by deep reinforcement learning. in *Proceedings of the 11th ACM SIGGRAPH Conference on Motion, Interaction and Games*. Limassol, Cyprus: Association for Computing Machinery
[26] Maeda, I. d., D.; Kitano, M.; Matsushima, H.; Sakaji, H.; Izumi, K.; Kato, A. (2020). Deep Reinforcement Learning in Agent Based Financial Market Simulation. *Journal of Risk and Financial Management.*, *13* (4), 71.
[27] Manríquez, F., Morales, N., Pinilla, G., and Piñeyro, I. (2019). Discrete event simulation to design open-pit mine production policy in the event of snowfall. *International Journal of Mining, Reclamation and Environment*, vol. *33* (8), pp. 572-588.

[28] McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, vol. *5, pp.* 115-133.
[29] Minkah, E. A. (2014). Determination of optimized Inchiniso pit, Ghana Bauxite Company Limited. MSc Thesis, University of Mines and Technology, Tarkwa, Pages 86.
[30] Ozdemir, B. and Kumral, M. (2018). Appraising production targets through agent-based Petri net simulation of material handling systems in open pit mines. vol. *87,* pp. 138-154.
[31] Ramazan, S. and Dimitrakopoulos, R. (2013). Production scheduling with uncertain supply: a new solution to the open pit mining problem. *Optimization and Engineering*, *14* (2), 361-380.
[32] Samuel, A. L. (1962). Artificial intelligence: a frontier of automation. *The Annals of the American Academy of Political and Social Science*, vol. *340* (1), pp. 10-20.
[33] Shishvan, M. S. and Benndorf, J. (2019). Simulation-based optimization approach for material dispatching in continuous mining systems. *European Journal of Operational Research*, *275* (3), 1108-1125.
[34] Shiv Prakash, U. and Hooman, A.-N. (2018). Simulation and optimization approach for uncertainty-based short-term planning in open pit mines. *International journal of mining science and technology*, vol. *28* (2), pp. 153-166.
[35] Silver, D. (2009). Reinforcement learning and simulation-based search in computer Go.Thesis, University of Alberta.
[36] Systèmes, D. (2019). Geovia Whittle. Strategic mine planning software. Ver. 4.7.3, Vancouver. Ver.
[37] Systèmes, D. (2020). Geovia GEMS. Strategic mine planning software. Ver. 6.8.3, Vancouver. Ver.
[38] Train, K. E. (2009). *Discrete choice methods with simulation.* Cambridge university press,
[39] Turing, A. (1950). Computing machinery and intelligence. *Parsing the Turing Test*, 23-65.
[40] Vasquez-Coronado, P. P. (2014). Optimization of the haulage cycle model for open pit mining using a discrete-event simulator and a context-based alert system. MSc Thesis, The University of Arizona, Tucson, Pages 156.
[41] Vidhya, A. (2020). Reinforcement Learning via Markov Decision Process. .
[42] Wiki, A. (2021). Gradient Descent. Retrieved from: https://docs.paperspace.com/machine-learning/wiki/gradient-descent
[43] Wiki, A. (2021). Weights and Biases. Retrieved from: https://docs.wandb.ai/
[44] Xie, J., Ge, F., Cui, T., and Wang, X. (2022). A virtual test and evaluation method for fully mechanized mining production system with different smart levels. *International Journal of Coal Science & Technology*, vol. *9* (1), pp. 41.