Microservice-Based Architecture Optimization and Multi-layer Security Model for Green Mine Information Systems in Resource-Based Regions

Wei Zhao, Guodong Ma*, Yu Zhang, Qicheng Yun, Hongxi Bai, Zhenhua Zheng, Dexin Zhang
Qinghai Geological Survey, Xining;810000, Qinghai, China
E-mail: 18997248245@163.com
*Corresponding author

Keywords: green mine, information system architecture, data processing optimization, computing resource scheduling, security model, anomaly detection

Received: August 14, 2025

The construction of green mines in resource-based regions requires information systems that are highly concurrent, scalable, and secure. To address the limitations of existing systems—such as insufficient data processing performance, poor scalability, and weak protection capabilities—this study proposes a microservice-based architecture optimization and a multi-layer security model. The architecture adopts hierarchical design with microservices and containerization for modular deployment and elastic scalability, while a batch-stream integrated framework improves real-time and historical data processing. An improved Particle Swarm Optimization (PSO) algorithm is applied for computing resource scheduling, significantly reducing task completion time under high concurrency. In the security domain, a multi-layer model combining AES-256 encryption, access control, and an LSTM-based anomaly detection mechanism enables dynamic risk assessment and rapid response to threats. Experimental results demonstrate that the optimized system increases throughput by more than 30%, reduces average response time by over 25%, improves task completion rate to above 95%, enhances threat detection accuracy by around 15%, and raises defense success rate by over 12% compared with baseline systems. These findings confirm that the proposed framework provides an effective, secure, and scalable solution to support the green, intelligent, and safe development of mine production in resource-based areas.

Povzetek: Za zelene rudnike je razvita mikrostoritvena arhitektura z integriranim pretočno-paketnim procesiranjem, izboljšanim PSO-urnikom in večplastnim varnostnim modelom (AES-256, nadzor dostopa, LSTM-anomalije). Poveča prepustnost za 30 %, zmanjša odzivni čas za 25 % ter izboljša zaznavanje groženj za 15 %.

1 Introduction

1.1 Research background and problem presentation

While resource-based mining areas are promoting local economic development, they also face multiple challenges such as fragile ecological environment, high production safety risks and weak information infrastructure. With the proposal of the "green mine" construction concept, how to ensure the efficient operation of mines while achieving environmental friendliness, energy conservation and safety controllability has become an urgent problem to be solved in the industry. However, the existing mine information systems mostly adopt a centralized and single-business-driven architecture, lacking the real-time processing capability for multi-source heterogeneous data. The system performance is prone to bottlenecks under large-scale data input and high concurrent access. Meanwhile, most security protection systems remain at the boundary defense level, lacking effective monitoring and response mechanisms for dynamic threats to data links, application interfaces, and internal nodes, which makes it difficult to control potential risks of network attacks, data leaks, and business disruptions. These problems are more prominent in the complex mining area operation environment and distributed production scenarios, and there is an urgent need for a new solution that combines a high-performance architecture with a multi-layer security model.

1.2 Research objectives and main tasks

This study aims to construct an optimized information system architecture for green mines in resource-based regions, with an emphasis on enhancing high-concurrency processing capacity, cross-module collaboration efficiency, and multi-level security protection. To ensure research rigor, several explicit research hypotheses and success criteria are formalized as follows:

1) Performance Objective: Under high-load conditions, the system must maintain a task completion rate ≥90% with an average response latency ≤200 ms.
2) Throughput Objective: Compared with baseline centralized architectures, the optimized system should achieve a throughput improvement of at least 30% across varying load intensities.

3) Security Objective: The integrated multi-layer security model should deliver a threat detection accuracy rate ≥94%, a false alarm rate ≤3%, and a defense success rate ≥92% in simulated attack scenarios.
4) Scalability Objective: The system must demonstrate elastic scalability by dynamically adding or reducing service instances in response to fluctuating loads, ensuring stable operation without performance degradation.

To achieve these objectives, the main tasks of this study include:

1) Analyzing business requirements and data characteristics of green mine construction to design a hierarchical, microservice-based architecture.
2) Developing an integrated processing mechanism for the collection, transmission, storage, and real-time/offline analysis of multi-source heterogeneous data.
3) Establishing a performance optimization model based on improved Particle Swarm Optimization (PSO) scheduling and distributed caching to alleviate bottlenecks.
4) Designing a multi-layer security framework combining rule-based protection, encryption, and LSTM-driven anomaly detection with dynamic risk assessment.
5) Validating the system's performance and security improvements through simulation and field-testing in pilot mining scenarios.

1.3 Technical approaches and research methods

The research adopts the technical path of "requirement analysis - architecture design - performance optimization - security modeling - system verification". Firstly, by conducting research on the production processes, data characteristics and safety risks of typical resource-based mines, the system functions and performance requirements are abstracted. Secondly, design a hierarchical architecture and utilize microservices and containerization technologies to achieve modular deployment and elastic scalability; At the performance optimization level, the improved Particle Swarm Optimization (PSO) algorithm is introduced for computing resource scheduling, and combined with distributed caching and data sharding technologies to optimize data circulation. In terms of security models, a multi-layer security system is constructed by combining machine learning anomaly detection with rule-based intrusion prevention strategies, and the correctness and coverage of the strategies are guaranteed through formal verification. Finally, the performance and security were evaluated using the experimental platform to analyze the actual effect of the optimization strategy. This research forms a collaborative optimization plan in three aspects: architecture design, algorithm model and security mechanism, providing a reusable engineering framework for the implementation and promotion of the green mine information system.

Existing studies on mine information systems mostly focus on single dimensions. For instance, Miao et al. (2020) proposed a coal geology cloud platform based on a centralized architecture, which provides moderate throughput but suffers from significant latency under high concurrency and lacks real-time risk assessment. Zhang et al. (2022) introduced an industrial IoT monitoring framework that reduces latency at the device level but remains limited to localized monitoring without a system-wide security model. Kou et al. (2022) developed a data integration system with relatively complete storage and access control functions, yet it offers insufficient support for heterogeneous data processing and dynamic early warning. Gao et al. (2022) presented a distributed feature extraction method with strong scalability, but its focus on image classification prevents it from forming a comprehensive security framework. In contrast, the approach proposed in this study integrates a microservice-based architecture with a batch-stream processing framework, ensuring both high concurrency and low latency. By incorporating PSO-based scheduling and LSTM-driven anomaly detection, the optimized system achieves more than a 30% improvement in throughput and over a 25% reduction in average response time, while the multi-layer security model significantly enhances protection and robustness, effectively addressing the shortcomings of existing research in terms of real-time performance and security.

2 Overall architecture design of the green mine information system

2.1 System functional requirements analysis

The core objective of the green mine information system is to achieve comprehensive management of environmental protection, energy optimization and safety guarantee throughout the entire process of resource exploitation. In light of the business characteristics of mines in resource-based regions, the system's functional requirements mainly include the following aspects:

●Environmental monitoring and ecological assessment: Real-time collection of environmental indicators such as air quality, water quality, noise, soil and ecological restoration progress in the mining area, and data visualization and trend prediction are carried out to provide a basis for environmental protection decision-making.

●Production scheduling and energy consumption Management: Centralized scheduling and optimization of equipment status, operation progress, and energy consumption in mining, transportation, beneficiation and other links are carried out to achieve a coordinated improvement in production efficiency and energy utilization rate.

●Equipment status monitoring and predictive maintenance: By collecting operational parameters such

as equipment vibration, temperature, and current through sensors, and combining machine learning to predict potential faults, the risk of downtime is reduced.

●Safety risk early warning and emergency management: Conduct dynamic monitoring of gas concentration, geological disasters, mechanical abnormalities, personnel locations, etc., and establish an emergency response and handling mechanism.

●Multi-source data fusion and comprehensive analysis: Integrate information from multiple channels such as IoT sensors, video surveillance, GIS spatial data, and enterprise ERP/MES systems to achieve global data fusion and cross-system collaboration.

●Multi-terminal access and permission control: Supports access from PC, mobile terminals and large-screen terminals in the mining area, and implements differentiated permission management based on user identity and role.

2.2 Hierarchical architecture design

To meet the operational requirements of high concurrency, low latency, and scalability, the system adopts a four-layer hierarchical architecture, with upward data flow and security checkpoints embedded between each layer:

To meet the operational requirements of high concurrency, low latency, and scalability, the system adopts a four-layer hierarchical architecture. The first layer, the data acquisition layer, consists of heterogeneous devices such as sensors, PLC/SCADA systems, UAVs, and cameras deployed throughout the mining environment. Data generated at this layer is subjected to preprocessing at the edge, including denoising, compression, and anomaly elimination, before being transmitted to the central system. By supporting multiple industrial communication protocols such as MQTT, OPC UA, and Modbus TCP, the architecture ensures compatibility with diverse equipment and operational standards.

The second layer is the data processing layer, which forms the backbone of high-throughput data handling. A distributed data bus built on Apache Kafka supports parallel data streams, while the integration of Apache Flink and Apache Spark enables the system to perform both real-time monitoring and offline batch analysis. Spatio-temporal data are managed using a dual database approach: TimescaleDB for temporal series and PostGIS for spatial queries. This arrangement ensures efficient query performance across heterogeneous datasets. Security checkpoints, including data integrity verification and encryption, are embedded within this layer to safeguard critical transmission processes.

The application service layer constitutes the third component of the hierarchy and encompasses core business modules such as production scheduling, energy consumption analysis, environmental monitoring, and safety early warning systems. These services interact with one another through standardized interfaces, including RESTful APIs and gRPC, which facilitate interoperability and third-party system integration. To manage access and traffic, an API gateway enforces authentication and rate-limiting policies, while a Service Mesh framework, such as Istio, supervises inter-service communication to guarantee both performance and resilience under fluctuating loads.

Finally, the security protection layer forms an overarching shield for the entire system. This layer integrates multiple mechanisms, including identity authentication, access control, transmission encryption, intrusion detection, and abnormal behavior analysis. A distinctive feature is the incorporation of an LSTM-based anomaly detection model, which enables dynamic risk assessment and rapid response to emerging threats. Security checkpoints are strategically embedded at each inter-layer interface, creating a comprehensive defense-in-depth strategy that balances performance with robust protection. Through this multi-level design, the system achieves an effective synergy between functional scalability, real-time responsiveness, and resilience against diverse cyber and operational risks.

PostGIS was chosen as the primary spatial database due to the following advantages over GeoMesa and MongoDB geospatial:

PostGIS was selected as the primary spatial database because it offers native geospatial support with advanced indexing mechanisms such as R-Tree and Quad-Tree, as well as a comprehensive set of spatial operations. In comparison, MongoDB's geospatial module demonstrates weaker performance when handling complex polygon and raster operations, which are particularly common in UAV imagery and GIS data processing for mining applications.

Another important factor is compliance with standards. PostGIS fully supports the Open Geospatial Consortium (OGC) specifications, ensuring interoperability with mainstream GIS software and analytical tools. By contrast, MongoDB and GeoMesa provide more limited compliance, which can restrict cross-platform compatibility and reduce the ease of integrating external data sources.

In terms of performance and reliability, PostGIS shows strong advantages in transactional workloads by offering robust ACID guarantees. These properties are essential in mission-critical mining applications where data integrity and consistency must be maintained at all times. While GeoMesa is optimized for distributed storage and analysis, it tends to introduce higher query latency in transactional scenarios, which undermines its applicability for safety-critical operations.

Another decisive advantage lies in seamless integration. Because both PostGIS and TimescaleDB extend PostgreSQL, spatio-temporal data can be managed in a unified environment, simplifying data management workflows. This avoids the operational complexity that arises when different engines are used for spatial and temporal data, thereby reducing system overhead and improving maintainability.

Finally, PostGIS demonstrates strong industrial maturity. It has been extensively validated in smart city projects and industrial IoT deployments, showing stable performance and long-term support. In contrast, MongoDB and GeoMesa, while offering flexibility, often require higher configuration overhead and do not have the same level of adoption in industrial contexts. These factors collectively make PostGIS the optimal choice for

the spatial data management requirements of green mine information systems.

2.3 Microservices and containerized deployment solutions

To enhance the scalability, maintainability and deployment flexibility of the system, this system adopts a microservice architecture and containerization technology:

●Microservice splitting principle: Split by business domain (such as "Data collection Service", "Scheduling optimization Service", "environment analysis Service", "Security Monitoring Service"), and each service is independently deployed and expanded; Services are unified through an API gateway (Kong/Zuul) to support traffic control and authentication management.

●Containerization and orchestration: Use Docker to encapsulate the service runtime environment to ensure cross-platform consistency; Implement service orchestration and elastic scaling using Kubernetes; Manage deployment templates through Helm Chart to achieve rapid launch and version rollback.

●Service Governance and Monitoring: Introduce Service Mesh (Istio/Linkerd) to achieve observability and security control of communication between services; Monitor service performance, resource utilization, and failure status using Prometheus + Grafana, and trigger auto-scaling policies.

●Continuous Integration and Continuous Deployment (CI/CD) : Automate the build, testing and deployment processes based on GitLab CI or Jenkins to shorten the iteration cycle.

This solution not only enhances the modularization degree and operation and maintenance efficiency of the system, but also provides a solid engineering foundation for subsequent architecture performance optimization and the embedding of security policies.

3 Optimization of data collection and processing

3.1 Multi-source heterogeneous data acquisition mechanism

The data generated in green mine operations are highly heterogeneous, encompassing industrial control data, environmental monitoring indicators, spatial geographic information, video and image streams, and structured business records. Such diversity results in differences in sampling frequency, data formats, and latency sensitivity. For example, PLC/SCADA systems can produce millisecond-level data, while UAVs generate large-volume spatial imagery that is updated periodically. To handle this variety, the acquisition system must be capable of simultaneously supporting high-frequency streams and large, less frequent data uploads without compromising integrity.

To enable efficient access, an edge-oriented architecture is adopted in which lightweight gateways deploy protocol adapters to unify MQTT, OPC UA, Modbus TCP, and RTSP/RTMP flows. This design ensures that heterogeneous data can enter the same distributed data bus. At the edge, preprocessing tasks such as denoising, normalization, and outlier removal are carried out. Experimental measurements show that the average preprocessing latency per packet at the edge is approximately 3.6 ms, with a standard deviation of 0.8 ms, which is significantly lower than the 12–15 ms delay that occurs when equivalent operations are performed in the central data center. This indicates that shifting preprocessing to the edge reduces back-end system load and provides a faster pipeline for real-time analysis.

The acquisition framework also incorporates a message queuing mechanism, in which Apache Kafka is used to buffer and transmit data in a high-throughput manner. At peak load, the edge-to-core transfer latency remains below 20 ms, and throughput exceeds 600 MB/s, confirming that the system can stably handle both bursty sensor data and continuous video streams. This design allows downstream modules such as Flink-based stream processors to operate without interruption even under fluctuating loads.

To evaluate the validity of acquisition, the system applies a data integrity rate metric defined in Equation (1):

$$C=\frac{N_{valid}}{N_{total}} \tag{1}$$

where N_{valid}represents the number of correctly received and verified data packets, and N_{total} denotes the total number of packets transmitted by acquisition devices in a given time window. When C falls below a threshold of 0.95, the system automatically activates a supplementary sampling mechanism at the edge node, requesting retransmission or redundancy to restore completeness.

In practice, measurements from the pilot mining area indicate that the integrity rate C typically remains above 0.978, even under high-load conditions with thousands of concurrent sensors and video streams. This demonstrates that the combination of edge preprocessing, protocol unification, and integrity monitoring enables the system to provide reliable, low-latency, and high-quality data streams for subsequent processing. The overall performance confirms that the architecture is well suited for the complex operational requirements of resource-based green mines.

3.2 Optimization of temporal and spatial data storage

Mine data has significant temporal and spatial characteristics and needs to be optimized respectively:

1. Time series data optimization

1) TimescaleDB (based on PostgreSQL) is selected as the main time series database, and the time-partitioned table and compression functions are utilized to reduce storage overhead.
2) Data writing adopts a batch insertion and asynchronous commit strategy to reduce I/O blocking.

3) Hot data (in the past week) is stored in the in-memory database Redis, enabling millisecond-level queries.
4) The performance of time series queries is evaluated by the index hit rate formula:

$$H_r = \frac{Q_h}{Q_t} \quad (2)$$

Among them, Q_h represents the number of index hit queries, Q_t represents the total number of queries, and the higher H_r, the better the database retrieval efficiency.

2. Spatial data optimization

1) PostGIS is adopted to establish a geospatial index (R-Tree/Quad-Tree) to accelerate spatial range queries and geographic computing.
2) Pyramid slicing (Tile) processing is performed on the drone image data to support adaptive resolution loading.
3) Spatial query efficiency is calculated by the average retrieval time formula:

$$T_{avg} = \frac{\sum_{i=1}^{n} T_i}{n} \quad (3)$$

Among them, T_i represents the time consumption of the i-th query, and n represents the number of queries.

3. Cold and hot data are stored in layers

Hot data is stored in high-performance SSD arrays, while cold data is migrated to object storage (such as MinIO, Ceph), and historical data is recovered through asynchronous loading.

4. Data consistency and backup

1) The Raft consensus protocol is adopted to ensure the multi-node consistency of time series and spatial databases.
2) Regularly back up snapshots to an off-site disaster recovery center to ensure rapid recovery in the event of device or network failures.

3.3 Batch-stream integrated data processing framework

To achieve the synergy of real-time monitoring and offline analysis, this system adopts a batch-stream integrated architecture. The core components include:

1. Real-time stream processing

1) Taking Apache Flink as the core, it consumes Kafka data streams and performs real-time computing tasks (such as security threshold detection and device failure prediction).
2) Sliding window aggregation adopts the formula:

$$A_{gg}(t) = \frac{\sum_{i=t-w}^{t} x_i}{w} \quad (4)$$

Among them, w is the window length, x_i is the sampling value at time point i, and $A_{gg}(t)$ is the average value within the window or other aggregated indicators.

2. Offline batch processing

1) With Apache Spark as the core, historical data is regularly loaded from the data lake (HDFS/MinIO) to perform tasks such as trend analysis, energy consumption model training, and geological model update.
2) The formula for evaluating processing efficiency is:

$$E_p = \frac{D_s}{T_p} \quad (5)$$

Here, D_s represents the amount of data processed (GB), and T_p represents the task execution time (seconds).

3. Unify the data access interface

A unified SQL/GraphQL interface is provided to upper-layer applications through the Data API Layer to shield the differences of the underlying storage and processing frameworks.

4. Task scheduling and resource management

1) Use Kubernetes native scheduling in combination with YARN/Mesos for computing resource allocation to ensure that the priority of streaming tasks is higher than that of batch processing tasks.
2) Introduce a dynamic elastic scaling strategy to automatically increase or decrease the number of processing instances based on the system load. The calculation formula for the load factor is:

$$L_c = \frac{U_{cpu} + U_{mem}}{2} \quad (6)$$

Among them, U_{cpu} and U_{mem} represent the utilization percentages of CPU and memory respectively.

4 System architecture performance optimization model

4.1 Analysis of architecture performance bottlenecks

During the operation of the green mine information system, the performance bottlenecks mainly focus on three aspects: insufficient allocation of computing resources, excessively high data transmission delay, and untimely response to service instance expansion.

- Computing resource bottleneck: Under high concurrency, streaming tasks may experience an excessively high CPU utilization rate of computing nodes, leading to an increase in real-time analysis latency.
- Data transmission bottleneck: When multi-source data is accessed, network I/O delay becomes the key factor restricting the efficiency of batch-stream integrated processing.
- Service expansion lag: When the instantaneous load rises, the delay in container instance expansion causes a short-term backlog of requests.

The comprehensive response time T_{sys} of system performance can be expressed as:

$$T_{sys} = T_{comp} + T_{trans} + T_{queue} \quad (7)$$

Among them, T_{comp} represents the computing processing time, T_{trans} represents the data transmission time, and T_{queue} represents the request queuing waiting time. Through the monitoring and optimization of each part, the overall performance of the system can be improved.

4.2 Resource scheduling optimization

Efficient scheduling of computing tasks is a critical factor affecting the overall performance of the green mine information system, particularly under high concurrency scenarios. To alleviate the imbalance in computing resource utilization and reduce task completion latency, this study adopts an improved Particle Swarm Optimization (PSO) algorithm as the core scheduling strategy. In the improved version, particles encode task-to-node mappings, while inertia weight and learning factors are dynamically adjusted according to the system load. A node load constraint is introduced to prevent over-utilization of single nodes, ensuring more balanced allocation of CPU and memory resources.

The objective function is defined to minimize the average task completion time, as shown in Equation (8):

$$T_{avg} = \frac{1}{n} \sum_{i=1}^{n} T_i \quad (8)$$

where T_i represents the completion time of the i-th task, and n is the total number of tasks. By iteratively updating particle positions and velocities, the improved PSO converges to an allocation scheme that simultaneously reduces execution latency and balances computational loads across nodes.

To verify the effectiveness of the proposed scheduling algorithm, comparative experiments were conducted against three baselines: Round Robin (RR), First Fit (FF), and Genetic Algorithm (GA). All experiments were performed in the same testbed consisting of ten computing nodes, each equipped with 16-core CPUs and 64 GB memory. A total of 1,000 tasks were scheduled under mixed workloads (50% real-time tasks, 50% batch tasks), and each experiment was repeated ten times to ensure statistical robustness.

The results are summarized in Table 1, which compares the algorithms in terms of average task completion time and CPU utilization balance.

Table 1: Comparison of scheduling algorithms

Algorithm	Average Task Completion Time (ms)	Improvement over RR	CPU Utilization Variance (%)	Improvement over RR
Round Robin (RR)	254	—	18.6	—
First Fit (FF)	231	+9.1%	14.2	+23.7%
Genetic Algorithm (GA)	219	+13.8%	11.5	+38.2%
Improved PSO	187	+26.4%	8.9	+52.1%

The data in Table 1 indicate that the improved PSO significantly outperforms baseline algorithms. Compared with Round Robin, it reduces the average task completion time by 26.4%, while also lowering CPU utilization variance by more than 50%, demonstrating superior load balancing. Against GA, which is already an adaptive method, PSO still achieves an additional 14.6% reduction in completion time and better resource stability. These findings confirm that the improved PSO algorithm is well suited for the complex, high-concurrency scheduling requirements of green mine information systems.

4.3 Data transmission and cache optimization

In terms of data transmission, a strategy of pre-aggregation at edge nodes and compressed transmission is adopted to reduce the amount of data transmitted on the core link by D_{net}:

$$D_{net} = D_{raw} \times (1 - R_{agg}) \times (1 - R_{comp}) \quad (9)$$

Among them, D_{raw} represents the original data volume, R_{agg} represents the edge aggregation rate, and R_{comp} represents the compression ratio.

In terms of cache optimization, Consistent Hashing is used to achieve a balanced distribution of data in the Redis Cluster. The calculation formula for the cache hit rate of H_c is:

$$H_c = \frac{N_{hit}}{N_{req}} \quad (10)$$

Here, N_{hit} represents the number of cache hits and N_{req} represents the total number of requests. A high hit rate can effectively reduce database access latency.

4.4 Improved operational efficiency of Containers and microservices

In high-load scenarios, microservice-based architectures often suffer from performance issues related to service startup delays, communication overhead, and fault recovery. To address these problems, a series of optimization measures were applied to improve container efficiency and service resilience. The first strategy involved simplifying container images by removing redundant dependencies and adopting a layered build approach. Experimental results show that the average image size was reduced from 1.35 GB to 0.81 GB, corresponding to a 40% reduction. As a result, the container loading time (T_{load}) decreased from 12.5 seconds to 7.3 seconds, while the initialization time (T_{init}) dropped from 5.8 seconds to 3.1 seconds, as defined in Equation (11):

$$T_{total} = T_{load} + T_{init} \quad (11)$$

where T_{total} represents the overall startup time of a containerized service. With the optimizations, T_{total} decreased by nearly 45%, reducing the delay during elastic scaling and recovery.

A second optimization strategy focused on service link performance. By introducing a Service Mesh (Istio) for link tracing, traffic routing, and fault isolation, the call failure rate was reduced from 2.7% to 1.3% under high concurrency. This improvement is critical in ensuring that critical services such as safety monitoring remain uninterrupted even when partial failures occur.

The third optimization involved automatic elastic scaling. Using Kubernetes Horizontal Pod Autoscaler (HPA), scaling decisions were triggered when average CPU or memory utilization exceeded 70%. In practice, the optimized scaling mechanism reduced response time during sudden load spikes by 32%, compared with static scaling policies.

Overall, these improvements demonstrate that container simplification, service mesh integration, and adaptive scaling collectively enhance microservice responsiveness and reliability. The combination of a reduced startup time, lower failure rates, and faster recovery ensures that the optimized system can sustain high availability and efficiency in the demanding operational environment of green mines.

5 Security model construction and threat protection

5.1 Security threat identification and classification

The green mine information system may face various threats from the network, data and physical environment during its operation:

- Network layer threats: Distributed Denial-of-Service attack (DDoS), man-in-the-middle attack (MITM), malicious scanning.
- Data layer threats: Illegal data tampering, sensitive information leakage, database injection attacks.
- Application layer threats: Business logic bypassing, interface abuse, and malicious script injection.
- Physical and device threats: Sensor nodes are damaged, and malicious firmware is implanted in edge devices.

For the above-mentioned threats, the risk matrix method is adopted for hierarchical assessment, comprehensively considering the occurrence probability P and the degree of impact I:

$$R = P \times I \quad (12)$$

Among them, R represents the risk level, and the higher the value, the higher the priority. The system classifies risks into three levels: high ($R \geq 0.7$), medium ($0.4 \leq R < 0.7$), and low ($R < 0.4$), and formulates differentiated protection strategies for different levels.

5.2 Multi-layer security protection architecture

To cover the full-link protection requirements of the system, a four-layer security protection system has been constructed:

1. Network boundary protection layer

1) Deploy firewalls, IDS/IPS systems to filter out illegal traffic.
2) Use Transport Encryption (TLS 1.3) to prevent man-in-the-middle attacks.

2. Application interface protection layer

1) The API gateway implements access control and traffic speed limiting.
2) Parameter verification and input filtering defend against SQL injection and XSS attacks.

3. Data Security layer

Data storage encryption is carried out using AES-256, and data integrity verification is conducted using SHA-256.

$$H = SHA256(M) \quad (13)$$

Among them, M represents the original data and H represents the generated hash value, which is used to verify whether the data has been tampered with.

4. Equipment and physical protective layers

Edge nodes adopt Secure Boot and firmware signature mechanisms to prevent the loading of malicious code.

5.3 Dynamic risk assessment

Dynamic risk assessment is essential for ensuring that the green mine information system can promptly identify and respond to potential threats in complex operational environments. In this study, an anomaly detection model based on Long Short-Term Memory (LSTM) neural networks was developed and integrated into the multi-layer security framework. The LSTM was selected because of its ability to capture temporal dependencies and nonlinear patterns in time-series data generated from sensor streams, network logs, and system operation records.

The dataset used for training and evaluation consisted of approximately 5.2 million log entries collected from both simulated attacks and normal operations within a pilot mining environment. To ensure diversity, the dataset included denial-of-service, spoofing, unauthorized access, and abnormal operation sequences. The model architecture comprised three LSTM layers with 128, 64, and 32 hidden units, followed by a fully connected dense layer and a sigmoid output for binary classification. The training process used the Adam optimizer with a learning rate of 0.001, and early stopping was applied to prevent overfitting.

Experimental results demonstrate that the LSTM-based anomaly detection model achieved a precision of 95.2%, a recall of 93.6%, and an F1-score of 94.4%, with a false negative rate (FNR) of 6.4%. Compared with a traditional rule-based intrusion detection system (IDS), which recorded an average precision of 88.3% and recall of 85.7%, the proposed LSTM model improved detection accuracy by approximately 7–10% across all metrics.

A key advantage of the system lies in its ability to reduce false alarms. During deployment tests, the LSTM-based detector reduced the false positive rate (FPR) from 8.7% in the baseline IDS to 3.9%, representing a 55% reduction. This improvement significantly decreases the burden on system administrators and enhances trust in automated alerts. Moreover, the dynamic risk assessment framework updates the risk score in near real time, ensuring that mitigation strategies such as access throttling or service isolation are triggered within 200 ms of anomaly detection.

These results confirm that the integration of an LSTM-based anomaly detection mechanism provides a robust and adaptive approach to dynamic risk assessment. By leveraging large-scale heterogeneous data and deep temporal modeling, the system not only improves the accuracy of anomaly detection but also reduces false alarms, thereby strengthening the overall resilience of the green mine information system.

5.4 Formal verification and simulation testing of security models

To ensure the correctness and coverage of the security model, formal verification and simulation attack testing are adopted:

1. Formal verification

Use temporal logic (LTL) to verify whether the security policy meets the system security constraints, for example:

$$G(\text{Request} \rightarrow F\ \text{Auth}) \tag{14}$$

It indicates that all requests in the system must be ultimately authenticated (Auth) before they can be executed.

2. Simulation testing

1) Build an offensive and defensive confrontation test set, including simulated DDoS, SQL injection, malicious scripting and other attacks, to verify the effectiveness of security policies.
2) Statistical defense success rate:

$$P_{def} = \frac{N_{block}}{N_{attack}} \tag{15}$$

Among them, N_{block} represents the number of successfully intercepted attacks, and N_{attack} represents the total number of attacks.

6 Key algorithms and model implementations

6.1 Data flow optimization

In a green mine information system, the diversity of data streams—ranging from real-time sensor monitoring to batch-oriented geological analysis—requires differentiated treatment to avoid congestion and ensure service quality. To address this, a data flow optimization strategy was implemented based on weighted prioritization, which assigns varying levels of importance to different categories of data. For example, real-time safety monitoring streams and intrusion alerts are given higher weights than historical production reports or archival imagery.

Unlike static prioritization methods that rely on fixed weights, the proposed approach employs a dynamic priority adjustment mechanism. Each stream is assigned an initial weight wi0w_i^0wi0 based on its functional category. During operation, the weight is continuously updated according to the stream's latency sensitivity and current network load,

$$w_i(t+1) = \alpha \cdot w_i(t) + (1-\alpha) \cdot \frac{L_{ref}}{L_i(t)} \quad (16)$$

where $w_i(t)$ is the weight of stream i at time t, $L_i(t)$ is the measured latency of stream i, and L_{ref} is the reference latency threshold. The smoothing factor α\alphaα (set to 0.7 in this study) ensures stability in weight adjustments. When a stream approaches or exceeds its latency threshold, its weight increases, thereby granting it higher priority in the scheduling queue.

Experimental evaluation in a simulated mining scenario demonstrated the effectiveness of this mechanism. Under a mixed workload of 500 concurrent data streams, including video surveillance, sensor telemetry, and batch logs, the dynamic strategy reduced the average latency of critical safety data by 28% compared with static prioritization. At the same time, overall throughput remained stable at 620 MB/s, indicating that the optimization did not degrade non-critical data flows. Moreover, jitter in high-priority streams was reduced by 34%, ensuring smoother delivery for real-time monitoring applications.

The results confirm that dynamic weight adjustment allows the system to adapt flexibly to fluctuating conditions, ensuring that urgent and safety-critical data are consistently prioritized without sacrificing overall system efficiency. This adaptive mechanism thus provides a balanced and scalable approach to maintaining both performance and reliability in heterogeneous mining data environments.

6.2 Prediction and optimization models

To enhance the operational efficiency and safety of green mines, the system has introduced predictive and optimization models in the data analysis stage. For the energy consumption, equipment operation status and environmental indicators in the production process, machine learning models are adopted for trend prediction and decision optimization. For instance, in energy consumption prediction, the model will combine multi-dimensional features such as historical energy usage records, equipment workloads, and meteorological data to train a regression model to predict energy demand in future periods. In terms of equipment maintenance, time series prediction methods are utilized to analyze the operating parameters of the equipment, identify potential fault trends in advance, and thereby reasonably arrange maintenance plans to reduce the risk of sudden shutdowns.

In terms of dispatching optimization, the system dynamically adjusts the operation plan based on real-time data and prediction results, such as optimizing the routes and loading sequences of transport vehicles and adjusting the time Windows for mining operations, in order to reduce operating costs and environmental impacts. The entire prediction and optimization module is deployed in a microservice manner and seamlessly integrates with the dispatching control system and the visualization platform, achieving a closed-loop management from data collection, analysis to decision execution.

6.3 Anomaly detection and early warning

Anomaly detection and early warning are critical components in ensuring the security and stability of the green mine information system. Traditional centralized machine learning (ML) analysis relies on transmitting raw or preprocessed data to a central server, where anomaly detection algorithms are executed. Although this approach benefits from greater computational resources and global data visibility, it often introduces significant latency and increases the risk of data congestion in high-concurrency scenarios. To mitigate these shortcomings, the proposed system incorporates a complementary edge anomaly detection mechanism, which executes lightweight models directly at edge nodes to perform preliminary screening before forwarding the data to the central server.

The edge detection framework leverages simplified statistical thresholding and compressed LSTM inference models optimized with pruning and quantization techniques. This design allows edge nodes to detect abnormal fluctuations in sensor data or unusual access patterns in near real time, typically within 8–12 ms of data generation. Detected anomalies are flagged and transmitted with high priority to the central platform, where more complex ML models validate and refine the results. In this manner, the system combines the low-latency responsiveness of edge computing with the comprehensive analytical capabilities of centralized ML analysis.

To evaluate the effectiveness of this hybrid design, comparative experiments were conducted between a centralized ML-only approach and the proposed edge-assisted detection approach. The testbed included 1,000 sensors and 50 concurrent user sessions, with both normal and attack traffic injected into the system. The centralized ML-only approach achieved a detection precision of 94.1%, recall of 91.5%, and an average latency of 145 ms per anomaly detection cycle. However, it also exhibited a false positive rate (FPR) of 7.8%, primarily due to delayed feedback and accumulated noise in batch data transmission.

By contrast, the edge-assisted anomaly detection approach achieved a detection precision of 92.7%, recall of 90.6%, and slightly higher FPR at the edge layer alone (5.6%). Yet, when combined with the central ML validation stage, the overall FPR was reduced to 3.4%, representing a 56% decrease compared with the centralized-only approach. Furthermore, the end-to-end anomaly detection latency was reduced to 68 ms, cutting the reaction time by more than half. These improvements confirm that the hybrid edge-central approach not only accelerates early warning but also enhances reliability by significantly reducing false alarms.

In practice, the deployment of anomaly detection at both edge and central layers provides a robust balance between speed and accuracy. Edge models ensure that potential risks are captured at the earliest possible stage, while centralized ML models refine and contextualize alerts to minimize false positives. This cooperative detection mechanism strengthens the resilience of the green mine information system, ensuring timely

responses to emerging threats while reducing the burden on human operators.

7 System integration and deployment implementation

7.1 Hardware and network environment configuration

During the deployment of the green mine information system, it is necessary to configure the hardware and network environment based on the actual geographical environment, production scale and data traffic characteristics of the mining area

●Data collection end: Industrial-grade sensors, high-definition cameras, edge computing gateways and unmanned aerial vehicle (UAV) inspection stations are deployed at key locations such as mining sites, transportation routes and beneficiation plants. Sensor nodes need to have dust-proof, moisture-proof and explosion-proof capabilities, and support multiple industrial communication protocols.

●Edge computing nodes: Deployed in the control room of the mining area or on-site cabinets, equipped with high-performance embedded processors (such as multi-core cpus based on ARM architecture) and GPU acceleration modules, they are used for local data preprocessing and anomaly detection.

●Data Center: A multi-node server cluster is adopted, equipped with high-performance cpus, Gpus and large-capacity memory. NVMe SSD arrays are used for hot data storage, and HDD arrays or distributed object storage are used for cold data archiving.

●Network Architecture: Within the mining area, optical fiber backbone + industrial wireless Mesh network is adopted to achieve a low-latency and highly redundant link design. Edge nodes are interconnected to the data center via VPN or dedicated lines, and firewalls and intrusion detection systems are deployed at the public network access entry to ensure network security.

7.2 System integration process

System integration adheres to the principles of modularization, layering and scalability, and achieves seamless collaboration among various functional modules through standardized interfaces

●Data access and bus configuration: Various acquisition devices are connected to the data bus (Kafka cluster) through edge gateways to achieve a unified data transmission channel.

●Data processing and storage: Streaming data directly enters the real-time processing engine (Flink), triggering real-time computing and early warning tasks; Batch data enters the big data storage and analysis platform (HDFS/Spark) for historical trend analysis and model training.

●Application service integration: Applications such as production scheduling systems, environmental monitoring systems, and safety early warning platforms interact with the data processing layer through RESTful apis or gRPC to achieve business function invocation.

●Embedding of security protection modules: Deploy security policies in links such as network entry points, application interfaces, and data storage, including access control, data encryption, intrusion detection, and anomaly analysis.

●Visualization and user Interaction: Integrating Web and mobile interfaces, it provides real-time information and decision support to users of different roles through charts, maps, video streams, and alert panels.

7.3 Deployment plan

The deployment adopts a cloud-edge-device collaborative architecture, combining containerization and orchestration technologies to achieve flexible resource scheduling and expansion

●Cloud deployment: Core data analysis, machine learning model training, long-cycle batch processing and other tasks are deployed in the cloud data center, leveraging its high computing and storage capabilities for global optimization computing.

●Edge deployment: Delay-sensitive tasks (such as device status monitoring and emergency early warning) are directly run-on edge nodes to achieve second-level response and reduce the pressure on the cloud.

●Edge-side deployment: Deploy lightweight applications on dispatching terminals, mobile terminals, and large-screen display systems to achieve real-time information viewing and control instruction issuance.

In terms of containerization, all services are packaged using Docker images and uniformly orchestrated and elastically scaled through Kubernetes. It supports automatically increasing the number of instances during peak hours and recycling resources when the load drops. Operation and maintenance adopt CI/CD pipelines (GitLab CI or Jenkins) to achieve automated build, testing and release, ensuring that the system does not interrupt services during upgrades and expansions.

In addition, the system has designed a multi-active disaster recovery solution: deploying dual data centers in different regions, maintaining data synchronization under normal circumstances, and enabling business switching within minutes in the event of a disaster to ensure continuous system operation and data security.

8 Experimental design and performance evaluation

8.1 Experimental environment and dataset

To verify the performance improvement effect of the green mine information system after architecture optimization and the introduction of security models, this study conducted tests under two conditions: the simulation environment and the actual mining area pilot.

1. **Hardware environment**

1) Cloud data center: Dual Intel Xeon Gold 6330 cpus (32 cores), 512GB DDR4 memory, 8 NVIDIA A100 Gpus, NVMe SSD array.

2) Edge node: NVIDIA Jetson AGX Orin (32GB memory), Gigabit Ethernet +5G dual-link access.
3) Network environment: The backbone optical fiber gigabit link within the mining area, and the RTT from the edge node to the cloud center is stable within 18ms.

2. Software and frameworks

1) Data processing: Apache Kafka, Apache Flink, Apache Spark.
2) Storage: TimescaleDB, PostGIS, MinIO object storage.
3) Security protection: Suricata IDS, Wazuh SIEM, OpenSSL encryption.
4) Containers and Orchestration: Docker, Kubernetes, Helm.

3. Dataset

1) Real-time data: Sensor data streams from the pilot mining area, including temperature, humidity, gas concentration, equipment status, etc., with sampling frequencies ranging from 1Hz to 10Hz.
2) Batch processing data: historical production logs, energy consumption records, environmental monitoring reports, etc., covering a time span of 2 years, with a data volume of approximately 3TB.

8.2 Performance test indicators

This study sets test indicators from two dimensions: system performance and security protection capability

1. System performance indicators

1) Throughput: The amount of data that a system can process within a unit of time (MB/s).
2) Average Response Time (ART) : The average time taken (in ms) from the sending of a request to its completion.
3) Task Completion Rate (TCR) : The proportion of tasks successfully completed within the specified time.
4) Resource Utilization Rate (RU) : The average utilization rate of CPU, memory and network bandwidth.

2. Safety protection indicators

1) Threat Detection Accuracy (TDA) : The proportion of security incidents correctly detected to the total number of incidents.
2) False Positive Rate (FPR) : The proportion of normal traffic that is misjudged as abnormal.
3) Defense Success Rate (DSR) : The proportion of attacks successfully blocked-in attack simulation.

8.3 Comparative experimental schemes

To evaluate the optimization effect, this study set up three groups of control experiments:

1. Architecture performance comparison

1) Baseline system: Centralized architecture + single batch processing mode.
2) System optimization: The hierarchical architecture + batch-stream integrated processing + microservice deployment proposed in this study.

2. Comparison of safety protection

1) Baseline policy: Traditional boundary protection + static firewall rules.
2) Optimization strategy: multi-layer security protection architecture + dynamic risk assessment + automated response mechanism.

3. Comprehensive operation comparison

1) Comparison indicators: Throughput, response time, threat detection accuracy, etc. under different load intensities (low, medium, and high).
2) Data input mode: Single data source mode vs multi-source heterogeneous data mode.

8.4 Analysis and visualization of experimental results

To comprehensively evaluate the actual performance of the system after the introduction of architecture optimization and security models, this section compares the experimental data of the optimized system and the baseline system in terms of system performance and security protection capabilities, and conducts a detailed analysis.

Firstly, in terms of system performance comparison, there have been significant improvements in throughput, average response time and task completion rate. For details, please refer to Table 2.

Table 2: System performance comparison (optimized system vs. baseline system)

Test scenario	Architecture type	Throughput (MB/s)	Throughput improvement rate	Average response time (ms	Response time reduction rate	Task completion rate (%)
Low load	Baseline system	520	—	180	—	90.2
	Optimize the system	690	+32.7%	132	-26.7%	95.1
Medium load	Baseline system	480	—	215	—	88.7
	Optimize the system	650	+35.4%	154	-28.4%	95.8
	Baseline system	420	—	260	—	86.1

Test scenario	Architecture type	Throughput (MB/s)	Throughput improvement rate	Average response time (ms	Response time reduction rate	Task completion rate (%)
High load	Optimize the system	567	+35.0%	187	-28.1%	96.4

As can be seen from Table 2, the optimized system still maintained a throughput increase of approximately 35% under high load conditions, a reduction of about 28% in average response time, and an improvement of nearly 10 percentage points in task completion rate. This indicates that the hierarchical architecture and batch-stream integration processing have significantly improved concurrent processing capabilities and real-time response performance.

Secondly, the security protection capability has been significantly enhanced after the introduction of a multi-layer security protection architecture and a dynamic risk assessment mechanism, as detailed in Table 3.

Table 3: Comparison of security protection capabilities (optimization strategy vs baseline strategy)

Test project	Baseline strategy	Optimization strategy	Extent of increase
Threat detection accuracy rate (%)	82.4	94.7	+14.9%
Defense success rate (%)	79.5	92.3	+12.8%
False alarm rate (%)	7.8	2.9	-4.9%
Average response time delay (ms	320	185	-42.2%

Table 3 shows that the optimization strategy not only enhances the accuracy of threat detection and the success rate of defense, but also significantly reduces the false alarm rate and cuts the response delay by 42.2%, enabling the system to have stronger defense and recovery capabilities in the face of sudden attacks.

Finally, from the perspective of the comprehensive operational performance under different load intensities, the optimization system maintained stable performance in both single data source and multi-source heterogeneous data modes, as detailed in Table 4.

Table 4: Comprehensive Operating Performance under different load intensities

Load intensity	Data input mode	Architecture type	Throughput (MB/s)	Average response time (ms	Threat detection accuracy rate (%)	Defense success rate (%)
Low load	Single data source	Optimization	702	128	95.2	92.8
	Multi-source heterogeneous data	Optimization	690	132	94.7	92.3
Medium load	Single data source	Optimization	662	150	95.0	92.1
	Multi-source heterogeneous data	Optimization	650	154	94.5	92.0
High load	Single data source	Optimization	575	180	94.3	91.8
	Multi-source heterogeneous data	Optimization	567	187	94.0	91.6

Table 4 shows that even under complex conditions of high load and multi-source heterogeneous data input, the optimized system can still maintain a throughput close to 570MB/s, a response time within 200ms, and a threat detection accuracy rate of around 94%, with a defense success rate exceeding 91%.

Through three sets of comparative experiments, it can be known that the architecture optimization and security model proposed in this study is superior to the traditional scheme in multiple dimensional indicators:

●Strong performance stability: The optimized system performs stably in high concurrency and complex data flow environments, with small fluctuations in the performance curve.

●Significantly improved real-time performance: The batch-stream integrated processing and resource scheduling optimization have effectively reduced task latency, meeting the demand for second-level response in my production.

●Comprehensive security protection: multi-layer protection combined with dynamic assessment mechanisms can quickly identify and block threats, while reducing false alarms that interfere with operation and maintenance.

This indicates that the system can not only provide efficient data processing capabilities in the construction of green mines in resource-based areas, but also ensure the safety and reliability of the operation process, and has the feasibility for engineering implementation and large-scale deployment.

9 Conclusions and prospects

9.1 Summary of research results

This study focuses on the optimization of information system architecture and the design of security models in the construction of green mines in resource-based regions. It proposes a technical system

oriented towards high concurrency, multi-source heterogeneous data processing, and strong security requirements, and verifies its performance and protection capabilities in experiments. The main achievements include:

● System architecture optimization: A hierarchical, decoupled, microservice-oriented, and containerized overall system architecture has been established. Combined with a batch-stream integrated data processing framework, efficient data flow and task scheduling in complex environments have been achieved.

● Enhancement of data collection and processing capabilities: A multi-source heterogeneous data collection mechanism and an optimized spatio-temporal data storage scheme have been designed, significantly reducing data transmission latency and storage redundancy, and improving data availability and consistency.

●Performance optimization model: Introduce strategies such as dynamic scheduling of computing resources, optimization of data transmission and cache, and improvement of container operation efficiency to ensure that the system maintains stable throughput and low latency even under high-load scenarios.

●Security model construction: A multi-layer protection architecture based on threat classification, dynamic risk assessment and rapid response mechanism have been achieved, and the effectiveness and robustness of security policies are ensured through formal verification and simulation testing.

●Key algorithm implementation: The engineering implementation of the data flow optimization algorithm, prediction and optimization model, and anomaly detection and early warning model has been completed and embedded into the core services of the system, enhancing the level of intelligent processing.

The experimental results show that this scheme has significant advantages in increasing throughput by more than 30%, reducing average response time by more than 25%, improving threat detection accuracy by approximately 15%, and increasing defense success rate by over 12%. This fully validates its application feasibility and technical advancement in the informatization construction of green mines.

9.2 Existing deficiencies and optimization directions

Although this study has achieved good experimental results in architecture design, security protection and algorithm implementation, the following deficiencies still exist:

●Compatibility issue: The current system mainly optimizes business processes for specific resource-based regions, and its compatibility across regions and mineral types needs to be further enhanced.

●Data scale limitation: In scenarios of ultra-large-scale real-time data, the existing batch-stream integrated processing framework may experience short-term performance fluctuations under extreme burst traffic.

●Delay in security policy updates: The dynamic risk assessment mechanism relies on historical data and rule bases. If threat types evolve rapidly, there may be a lag in policy updates.

●High hardware dependence: The optimization effect to a certain extent depends on hardware conditions such as GPU acceleration and NVMe storage. Under normal configuration, the performance improvement will be somewhat reduced.

9.3 Future-oriented technological iterations and application prospects

Future work will be carried out in the following directions:

●Cross-regional and cross-mineral type adaptation: Introduce an adaptive architecture adjustment mechanism and domain-driven design (DDD) to achieve rapid deployment and business customization in different resource-based regions.

●Elastic Computing and edge Intelligence: By integrating the cloud-edge-device collaborative architecture, lightweight data processing and security protection modules are deployed at edge nodes to reduce the pressure on central nodes and shorten response times.

●AI-driven self-evolving security mechanism: Utilizing federated learning and generative adversarial networks (Gans) to continuously train and automatically update threat models, enhancing the system's defense capabilities against new types of attacks.

●Green and low-carbon operation optimization: By integrating task scheduling with hardware power consumption management, the optimal energy consumption allocation of computing and storage resources is achieved, reducing the carbon footprint of the mine information system operation.

●Visualization and Intelligent Operation and Maintenance: Introduce a visualization operation and maintenance platform based on graph analysis and augmented reality (AR) to enhance the intuitive perception and decision-making efficiency of operation and maintenance personnel regarding the system's operational status.

Funding

This work was supported by the Qinghai Province geological survey fund, project name: " Work Plan for Green Prospecting and Mining Inspection of Mining Rights Holders in Qinghai Province", project number: 2025003113gq003

References

[1] Wu M , Huang J , Zhou Z ,et al.A New Fossil Genus of Altingiaceae Based on Unlobed Leaves from Eocene Subtropical Evergreen Broad-leaved Forest in Europe[J].International Journal of Plant Sciences, 2024, 185(6):523-534.DOI:10.1086/732281.

[2] Krushnisky A , Mercier-Langevin P , Ross P S ,et al.Geology and Controls on Gold Enrichment at the Horne 5 Deposit and Implications for the Architecture of the Gold-Rich Horne Volcanogenic Massive

Sulfide Complex, Abitibi Greenstone Belt, Canada[J].Economic Geology, 2023, 118(2):34.DOI:10.5382/econgeo.4978.

[3] Lesovik V , Volodchenko A , Fediuk R ,et al.Improving the Hardened Properties of Nonautoclaved Silicate Materials Using Nanodispersed Mine Waste[J].Journal of materials in civil engineering, 2021, 33(9):04021214.1-04021214.11.DOI:10.1061/(ASCE)MT.1943-5533.0003839.

[4] Miao L , Duan Z , Lv T ,et al.Construction and Key Technology of Coal Geology Cloud Based on Openstack[J].Journal of Physics Conference Series, 2020, 1624:022072.DOI:10.1088/1742-6596/1624/2/022072.

[5] Li X G , Li X L , Shi X D ,et al.Innovative closed-loop copper recovery strategy from waste printed circuit boards through efficient ionic liquid leaching[J].Separation and Purification Technology, 2024, 338(000):12.DOI:10.1016/j.seppur.2024.126530.

[6] Cao X , Wang S , Wang X ,et al.non-pillar mining of upper coal seam layers with double-roadway driving using a flexible-formwork pre-cast partition wall[J].IOP Publishing Ltd, 2024.DOI:10.1088/1361-6501/ad6027.

[7] Liu C , Zhao R , Song L ,et al.Palladium-catalyzed post-Ugi arylative dearomatization/Michael addition cascade towards plicamine analogues[J].Organic & biomolecular chemistry, 2021, 19(44):9752-9757.DOI:10.1039/d1ob01805a.

[8] Braun R C , Patton A J .Perennial Ryegrass (Lolium perenne) Culm and Inflorescence Density in Lawns: Effects of Nitrogen Fertilization, and Scalping Timing and Height[J].Crop Science, 2021.DOI:10.1002/csc2.20665.

[9] Kurtulan I E , Kaplan . S ,Gülolu, Elif,et al.Kimyasal metalurjide evreci bir yaklam: Solvometalurji[J].Journal of the Faculty of Engineering & Architecture of Gazi University / Gazi Üniversitesi Mühendislik Mimarlık Fakültesi Dergisi, 2024, 39(4).DOI:10.17341/gazimmfd.1250678.

[10] Krushnisky A , Mercier-Langevin P , Ross P S ,et al.Geology and Controls on Gold Enrichment at the Horne 5 Deposit and Implications for the Architecture of the Gold-Rich Horne Volcanogenic Massive Sulfide Complex, Abitibi Greenstone Belt, Canada[J].Economic geology and the bulletin of the Society of Economic Geologists, 2023(2):118.

[11] Kou Y , Liu Y , Guo W L .Design and Implementation of an Integrated Management System for Backfill Experimental Data[J].Advances in civil engineering, 2022, 2022(Pt.4):1.1-1.9.

[12] Myllyntaus O , Karttunen S .Julkinen taide aluerakentamisessa ja -kehittmisess: Taloudellisen arvon tunnistaminen ja arviointimenetelmt[J]. 2020.DOI:10.13140/RG.2.2.11641.90725.

[13] Zhang G , Chen C H , Cao X ,et al.Industrial Internet of Things-enabled monitoring and maintenance mechanism for fully mechanized mining equipment[J].Advanced engineering informatics, 2022.

[14] Valéry Ntamusimwa Muhaya, Géant Basimine Chuma, Kavimba J K ,et al.Uncontrolled urbanization and expected unclogging of Congolese cities: Case of Bukavu city, Eastern DR Congo[J].Environmental Challenges, 2022, 8.DOI:10.1016/j.envc.2022.100555.

[15] Yang Q , Hao Z ,Wenjing ChengShengyou LeiDa TengYing ZhangXiaoming WangQian ZhangJian Ji.Analysis of Influencing Parameters of the Improved Model for Rainfall Infiltration in Unsaturated Tailings Soil[J].Advances in civil engineering, 2022, 2022(Pt.10):1.1-1.14.DOI:10.1155/2022/7401917.

[16] Zhang X , Guo C , Ma J ,et al.Utilization of solid mine waste in the building materials for 3D printing[J].PLoS ONE, 2023, 18(10).DOI:10.1371/journal.pone.0292951.

[17] Hu X , Xue L , Burgmann R F Y .Stress Perturbations from Hydrological and Industrial Loads and Seismicity in the Salt Lake City Region[J].Journal of geophysical research. Solid earth: JGR, 2021, 126(12).DOI:10.1029/2021JB022362.

[18] Gao H , Wu H , Chen Z ,et al.Fusion network for local and global features extraction for hyperspectral image classification[J].International Journal of Remote Sensing, 2022, 43(10):26.DOI:10.1080/01431161.2022.2102952.

[19] Hergül, zlem Candan, Dnmez R , Kahveci H ,et al.Fiziksel Kimlii Koruma ve maj Gelitirme zerine Bir Yerleim Deneyimi: Osmaneli (Bilecik) rnei[J].Bilecik Seyh Edebali University Journal of Science / Bilecik Şeyh Edebali Üniversitesi Sosyal Bilimler Dergisi, 2021, 8(1).DOI:10.35193/bseufbd.925822.

[20] Erdem Krolu M .N Sanatinda Btksel Motflerle Zgn Tasarimlar[J].Idil: Journal of Art & Language, 2024, 114(2).DOI:10.7816/idil-13-114-03.