



Optimización cuántica para planificación minera con capacidad flexible: Implementación de QAOA con variables de holgura

Quantum optimization for mine planning with flexible capacity: QAOA implementation with slack variables

Bairon Rojas-Castro¹, Aldo Quelopana¹

¹ Universidad Católica del Norte. Departamento de Ingeniería de Sistemas y Computación. Antofagasta, Chile.

DOI: 10.64966/ingeniare.v34.01

Resumen

Este trabajo presenta un estudio exploratorio y preliminar sobre la aplicación de un algoritmo cuántico variacional para resolver problemas de planificación minera a cielo abierto con capacidad de extracción flexible. El objetivo principal es evaluar la viabilidad técnica de la formulación propuesta y su resolución mediante QAOA, más que comparar su rendimiento frente a métodos clásicos o cubrir un amplio rango de instancias. Se desarrolla una formulación Quadratic Unconstrained Binary Optimization (QUBO) que incorpora variables de holgura para modelar restricciones de capacidad del tipo “menor o igual”, permitiendo operaciones por debajo de la capacidad máxima sin penalizaciones. La metodología emplea el Quantum Approximate Optimization Algorithm (QAOA) para optimizar la selección de bloques maximizando la ganancia económica de la extracción mientras respeta restricciones de precedencia geológica y límites operativos flexibles. El enfoque se valida mediante un caso de estudio en un yacimiento sintético de cobre de 18 bloques resuelto en un simulador cuántico, demostrando la capacidad del algoritmo cuántico para manejar la complejidad adicional introducida por las variables auxiliares. Esta investigación establece fundamentos para aplicar computación cuántica a problemas de planificación minera con mayor flexibilidad operativa.

Palabras clave: Computación cuántica, QAOA, QUBO, planificación minera.

Abstract

This work presents an exploratory, preliminary study of applying a variational quantum algorithm to solve open-pit mine planning problems with flexible extraction capacity. The main objective is to assess the technical feasibility of the proposed formulation and its resolution using QAOA, rather than to benchmark its performance against classical methods or address a wide range of instances. A Quadratic Unconstrained Binary Optimization (QUBO) formulation is developed that incorporates slack variables to model capacity constraints as “less than or equal to,” allowing operations below maximum capacity without penalties. The methodology employs the Quantum Approximate Optimization Algorithm (QAOA) to optimize block selection by maximizing the economic gain from extraction while respecting geological precedence constraints and flexible operational limits. The proposed approach is validated through a case study involving an 18-block synthetic copper deposit, which is solved using a quantum simulator. The results demonstrate the quantum algorithm’s capacity to address the increased complexity introduced by auxiliary variables. This study establishes a framework for applying quantum computing to mining planning problems that require more prominent operational flexibility.

Keywords: Quantum computing, QAOA, QUBO, mine planning.

INTRODUCCIÓN

La planificación minera estratégica constituye uno de los problemas de optimización más complejos en la industria extractiva, donde la determinación óptima de qué bloques extraer y en qué secuencia impacta directamente en la viabilidad económica de los proyectos [1]. Los modelos de planificación minera tradicionalmente emplean restricciones de capacidad del tipo "menor o igual" que permiten operar por debajo de la capacidad máxima, reconociendo la realidad operativa donde las minas rara vez funcionan exactamente a su capacidad nominal debido a factores como mantenimiento preventivo de equipos, variaciones en la dureza del material, condiciones climáticas adversas, y consideraciones de seguridad [2]. Esta flexibilidad operativa proporciona mayor robustez y permite mejor adaptación a las condiciones dinámicas del entorno minero [3].

Bajo el enfoque Direct Block Schedule (DBS), la planificación minera en operaciones a cielo abierto es un problema de la clase NP-hard en términos de complejidad computacional. El espacio de soluciones posibles experimenta un crecimiento exponencial en función del número de bloques considerados, y la literatura no reporta algoritmos de complejidad polinomial capaces de garantizar la optimalidad global en casos generales [4], [5].

Diversos enfoques clásicos han sido propuestos para abordar este problema, incluyendo métodos exactos basados en programación entera, metaheurísticas como Particle Swarm Optimization, Simulated Annealing, Tabu Search y Variable Neighborhood Descent, así como técnicas de inteligencia artificial basadas en redes neuronales y aprendizaje por refuerzo [6] - [13]. Aunque estas metodologías pueden producir soluciones de buena calidad, presentan limitaciones relevantes en escalabilidad, interpretabilidad, garantías de optimalidad y sensibilidad a la configuración de hiperparámetros [14], [15]. Estas restricciones motivan la necesidad de explorar enfoques alternativos capaces de manejar de manera más robusta la complejidad combinatoria inherente al problema.

En este contexto, la computación cuántica emerge como una tecnología prometedora para abordar la complejidad combinatoria de problemas de planificación minera con capacidad flexible, trabajando de manera híbrida con algoritmos clásicos [16]. Los algoritmos cuánticos variacionales como el Quantum Approximate Optimization Algorithm (QAOA) [17], particularmente, aprovechan la superposición cuántica y el entrelazamiento para explorar eficientemente espacios de solución de alta dimensionalidad [18]. Sin embargo, estos métodos requieren formulaciones Quadratic Unconstrained Binary Optimization (QUBO) [19] que incorporan restricciones de igualdad mediante términos de penalización cuadráticos [20].

El presente estudio desarrolla una formulación QUBO específica para problemas de planificación minera con restricciones de capacidad flexible, empleando variables de holgura para convertir restricciones de desigualdad en restricciones de igualdad manejables por algoritmos cuánticos. Esta formulación se resuelve mediante QAOA implementado en un simulador cuántico con Qiskit [21], demostrando la viabilidad técnica del enfoque cuántico para manejar la complejidad adicional introducida por la flexibilidad de capacidad. Esto representa un avance en comparación con enfoques cuánticos anteriores, los cuales no consideran capacidades flexibles y solamente consideran restricciones de precedencia de bloques [22] - [24].

Esta investigación forma parte de una línea de trabajo preliminar y exploratorio, orientado a verificar la viabilidad técnica de aplicar computación cuántica a problemas de planificación minera con capacidad flexible. El presente trabajo no busca aún realizar comparaciones exhaustivas con métodos clásicos ni evaluar múltiples instancias, sino establecer una base para futuros estudios con mayor alcance.

FUNDAMENTOS DE LA COMPUTACIÓN CUÁNTICA

La computación cuántica representa un paradigma computacional que explota principios de la mecánica cuántica para realizar cálculos que pueden superar las capacidades de sistemas clásicos en dominios específicos. Los qubits, unidades fundamentales de información cuántica, pueden existir en superposiciones coherentes de estados computacionales, permitiendo el procesamiento paralelo de múltiples configuraciones simultáneamente [18].

El desarrollo tecnológico actual ha producido múltiples implementaciones físicas de computadoras cuánticas, cada una optimizada para diferentes clases de problemas computacionales:

- **Sistemas Basados en Compuertas Cuánticas:** Utilizan secuencias de operaciones unitarias aplicadas a qubits individuales o conjuntos pequeños de qubits. Este modelo proporciona la máxima flexibilidad algorítmica y representa el estándar para computación cuántica de propósito general [25]. Los circuitos cuánticos en este paradigma manipulan estados mediante rotaciones controladas y operaciones de entrelazamiento.
- **Computación Cuántica Adiabática:** Resuelve problemas de optimización mediante evolución gradual desde un Hamiltoniano inicial simple hacia un Hamiltoniano objetivo que codifica el problema. Este enfoque es naturalmente adecuado para problemas de optimización combinatoria y ha inspirado el desarrollo de algoritmos como QAOA [26].
- **Arquitecturas Topológicas:** Explotan propiedades topológicas de ciertos sistemas cuánticos para proteger información contra errores. Aunque prometedoras para la tolerancia a fallos, estas tecnologías permanecen en etapas experimentales [27].
- **Sistemas de Variables Continuas:** Operan con grados de libertad continuos en lugar de qubits discretos, ofreciendo ventajas para simulación cuántica y ciertos problemas de optimización [28].

Las implementaciones cuánticas actuales, aunque han alcanzado escalas de más de 1,000 qubits físicos [29], enfrentan limitaciones fundamentales debido a decoherencia, errores de compuertas, y ausencia de corrección de errores robusta [30], [31]. Estas limitaciones restringen tanto el número de qubits lógicos efectivos como la profundidad de circuitos cuánticos implementables, sin embargo, aún con estas dificultades, la construcción de algoritmos cuánticos ha capturado el interés de gobiernos, instituciones académicas y empresas a nivel global, para prepararse a la llegada de hardware adecuado para estos [32].

Los simuladores cuánticos clásicos juegan un papel crucial en el desarrollo algorítmico actual [33]. Estas herramientas emulan el comportamiento cuántico usando recursos computacionales clásicos, facilitando el desarrollo y caracterización de algoritmos cuánticos antes de su implementación en hardware real. Aunque limitados por el crecimiento exponencial de recursos, los simuladores permiten la investigación algorítmica para sistemas de tamaño moderado [32].

La optimización cuántica aborda diferentes categorías de problemas según la naturaleza de sus variables de decisión. Para problemas de optimización discreta, la reformulación cuántica típicamente involucra la construcción de Hamiltonianos cuyos estados fundamentales codifican soluciones óptimas [34]. Esta aproximación es especialmente efectiva para problemas NP-hard donde métodos exactos clásicos resultan computacionalmente intratables.

Los problemas de planificación minera con variables de holgura encajan naturalmente en este framework, ya que tanto las variables de decisión principales (extraer/no extraer bloques) como las variables auxiliares (capacidad no utilizada) son inherentemente discretas. La codificación de estas variables en qubits permite aprovechar el paralelismo cuántico para explorar simultáneamente múltiples configuraciones de extracción y utilidades de capacidad.

Formulación QUBO

QUBO es una formulación matemática fundamental que representa problemas de optimización mediante variables binarias, donde las restricciones se incorporan al objetivo a través de términos de penalización [19]. Aunque QUBO puede expresar tanto problemas de maximización como de minimización, QAOA requiere que se formulen como problemas de minimización. Los problemas QUBO generan problemas de optimización combinatoria donde la complejidad crece cuadráticamente con el número de variables, aprovechando la propiedad característica de las variables binarias $x_i^2 = x_i$.

Un problema QUBO se expresa generalmente como se indica en la ecuación (1) y ecuación (2):

$$\min_x f(x) = x^T Q x = \sum_{i=1}^n \sum_{j=1}^n q_{i,j} x_i x_j \quad (1)$$

Sujeto a:

$$x_i \in \{0, 1\}, \forall i \in \{1, \dots, n\} \quad (2)$$

Donde:

- x es un vector de variables binarias de decisión,
- Q es una matriz real simétrica de tamaño $n \times n$,
- $q_{i,j}$ representa el costo asociado a la interacción entre las variables x_i y x_j ,
- n es el número de variables binarias.

La matriz Q permite codificar términos lineales y cuadráticos de la función objetivo, donde la diagonal $q_{i,i}$ representa coeficientes de términos lineales y los elementos fuera de la diagonal $q_{i,j}$ ($i \neq j$) representan términos cuadráticos. La simetría de la matriz permite escribir la ecuación (3):

$$f(x) = \sum_{i=1}^n q_{i,i} x_i + \sum_{i < j} q_{i,j} x_i x_j \quad (3)$$

Para problemas de planificación minera con variables de holgura, la matriz Q debe expandirse para acomodar tanto variables principales (decisiones de extracción de bloques) como variables auxiliares (representando capacidad no utilizada). Esta expansión incrementa la dimensionalidad, pero mantiene la estructura cuadrática fundamental que permite la implementación eficiente en algoritmos cuánticos variacionales.

Quantum approximate optimization algorithm (QAOA)

QAOA representa un algoritmo cuántico variacional híbrido especialmente diseñado para problemas de optimización combinatoria en la era de dispositivos cuánticos de escala intermedia [17]. Su arquitectura combina evolución cuántica controlada con optimización clásica de parámetros, proporcionando un enfoque práctico para problemas *NP-hard* como la planificación minera con capacidad flexible.

El algoritmo construye un estado cuántico parametrizado mediante la aplicación alternada de dos tipos de evolución: evolución bajo el Hamiltoniano de costo H_C , que codifica la función objetivo y penalizaciones por violación de restricciones, y evolución bajo el Hamiltoniano mezclador H_M , que facilita transiciones entre estados computacionales para exploración del espacio de soluciones.

La secuencia de operaciones se repite p veces, donde p representa la profundidad del circuito y controla la expresividad del *ansatz* cuántico. El estado resultante depende de parámetros variacionales $\vec{\gamma} = (\gamma_1, \dots, \gamma_p)$ y $\vec{\beta} = (\beta_1, \dots, \beta_p)$, cuya optimización se realiza mediante algoritmos clásicos que minimizan el valor esperado de la función de costo.

El proceso iterativo de QAOA se describe en el siguiente pseudocódigo:

Algoritmo 1: Quantum Approximate Optimization Algorithm (QAOA)

- 1: Función QAOA(H_C, p)
- 2: Inicializar parámetros $\vec{\gamma} = (\gamma_1, \gamma_2, \dots, \gamma_p)$ y $\vec{\beta} = (\beta_1, \beta_2, \dots, \beta_p)$
- 3: Preparar el estado inicial: $|\psi_0\rangle = |+\rangle^{\otimes n}$
- 4: Para $k = 1$ hasta p hacer:
 - 5: Aplicar operador de costo: $U_C(\gamma_k) = e^{-i\gamma_k H_C}$
 - 6: Aplicar operador mezclador: $U_M(\beta_k) = e^{-i\beta_k H_M}$,
 donde $H_M = \sum_{i=1}^n \sigma_i^x$
- 7: Fin Para
- //Estado *ansatz* resultante:

$$|\psi(\vec{\gamma}, \vec{\beta})\rangle = U_M(\beta_p)U_C(\gamma_p)\cdots U_M(\beta_1)U_C(\gamma_1)|\psi_0\rangle$$
- 8: Medir el estado cuántico para obtener la cadena de bits z
- 9: Evaluar la función de costo $H_C(z)$

10: Actualizar $\vec{\gamma}$ y $\vec{\beta}$ usando optimización clásica

11: Retornar z

12: Fin Función

Para problemas formulados como QUBO se requiere una transformación al Hamiltoniano de costo H_C , lo cual se logra expresar en términos de operadores Pauli. La conversión se basa en el mapeo directo entre variables binarias clásicas $x_i \in \{0, 1\}$ y operadores Pauli-Z mediante la relación que se indica en la ecuación (4):

$$x_i = \frac{1 - \sigma_i^z}{2} \quad (4)$$

Donde σ_i^z es el operador Pauli-Z actuando sobre el qubit i, con valores propios $\{+1, -1\}$ correspondientes a los estados computacionales $|0\rangle$ y $|1\rangle$ respectivamente.

Aplicando esta transformación a la formulación QUBO general de la ecuación (3), se obtiene la ecuación (5):

$$f(x) = \sum_{i=1}^n q_{i,i} \frac{1 - \sigma_i^z}{2} + \sum_{i < j} q_{i,j} \frac{1 - \sigma_i^z}{2} \frac{1 - \sigma_j^z}{2} \quad (5)$$

Expandiendo los términos cuadráticos y reagrupando, el Hamiltoniano de costo resultante adopta la forma que observa en la ecuación (6):

$$H_C = \sum_{i=1}^n h_i \sigma_i^z + \sum_{i < j} J_{i,j} \sigma_i^z \sigma_j^z + \text{constante} \quad (6)$$

Donde los coeficientes transformados siguen la ecuación (7) y ecuación (8):

$$h_i = -\frac{1}{2} q_{i,i} - \frac{1}{4} \sum_{j \neq i} q_{i,j} \quad (7)$$

$$J_{i,j} = \frac{q_{i,j}}{4} \quad (8)$$

Los coeficientes h_i capturan términos lineales (incluyendo contribuciones individuales de bloques y penalizaciones por violación de restricciones) y $J_{i,j}$ representan interacciones cuadráticas entre variables. Esta formulación permite codificar naturalmente problemas con restricciones complejas mediante la incorporación de términos de penalización apropiados en los coeficientes de la matriz QUBO.

El Hamiltoniano mezclador H_M facilita la exploración del espacio de soluciones mediante transiciones uniformes entre todos los estados computacionales. Junto con el Hamiltoniano de costo,

define completamente el *ansatz* variacional de QAOA, determinando la forma funcional del estado cuántico parametrizado que aproxima la solución óptima.

El *ansatz* de QAOA se construye mediante la aplicación alternada de operadores de evolución derivados de ambos Hamiltonianos, como se observa en la ecuación (9):

$$|\psi(\vec{\gamma}, \vec{\beta})\rangle = U_M(\beta_p)U_C(\gamma_p)\cdots U_M(\beta_1)U_C(\gamma_1)|+\rangle^{\otimes n} \quad (9)$$

Donde $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ es el estado de superposición uniforme inicial. La elección del Hamiltoniano mezclador determina directamente la forma de los operadores $U_M(\beta_k) = e^{-i\beta_k H_M}$ y, por tanto, la expresividad del *ansatz*.

Para problemas de optimización binaria general, el Hamiltoniano mezclador más comúnmente utilizado (estándar) se muestra en la ecuación (10):

$$H_M = \sum_{i=1}^n \sigma_i^x \quad (10)$$

Donde σ_i^x son operadores Pauli-X que actúan sobre cada qubit individual. Este operador tiene la propiedad de "voltar" qubits individuales: $\sigma_x|0\rangle = |1\rangle$ y $\sigma_x|1\rangle = |0\rangle$, permitiendo transiciones locales entre estados computacionales.

En este trabajo utilizamos el *ansatz* estándar de QAOA, ya que las restricciones de precedencia y capacidad se incorporan mediante penalizaciones en el Hamiltoniano de costo, manteniendo la flexibilidad de exploración del *ansatz* estándar mientras se beneficia de su universalidad aproximada.

La presencia de variables de holgura incrementa la complejidad del paisaje de optimización, frecuentemente requiriendo mayor profundidad de circuito para alcanzar aproximaciones de alta calidad. La determinación óptima de p debe balancear expresividad del *ansatz* contra complejidad implementacional y susceptibilidad al ruido. Para problemas de planificación minera con capacidad flexible, valores típicos de p pueden ser mayores que los requeridos para formulaciones con restricciones exactas, debido a la necesidad de capturar interacciones complejas entre variables principales y auxiliares.

QAOA ofrece ventajas específicas para problemas de planificación minera con capacidad flexible: 1) capacidad natural para manejar la complejidad combinatoria incrementada, 2) flexibilidad para incorporar múltiples tipos de restricciones mediante diseño de Hamiltonianos apropiados, 3) escalabilidad teórica a medida que mejore la tecnología cuántica, y 4) compatibilidad con conocimiento específico del dominio minero mediante personalización de operadores.

PROBLEMA DE PLANIFICACIÓN MINERA

El problema de planificación minera a cielo abierto que aborda este trabajo se enfoca en la selección óptima de bloques para extracción bajo datos estimados (un sólo escenario). La formulación considera un modelo de bloques tridimensional donde cada bloque representa una unidad discreta del yacimiento con características geológicas y económicas específicas. La decisión de extraer o no extraer cada bloque debe respetar restricciones de estabilidad geotécnica que imponen un orden de precedencia, garantizando que no se pueden extraer bloques inferiores sin haber removido previamente los bloques superiores que los soportan.

Siendo:

- N el número de total de bloques en el modelo de bloque.
- V_i el valor económico neto del bloque i (puede ser positivo o negativo).
- P_i el conjunto de bloques que deben extraerse antes que bloque i .
- x_i la variable de decisión binaria, donde $x_i = 1$ si el bloque es extraído y $x_i = 0$ en caso contrario.

La ecuación (11) muestra la función objetivo del problema, la cual busca determinar qué bloques extraer para maximizar el beneficio total:

$$\max_x Z = \sum_{i=1}^N V_i x_i \quad (11)$$

Sujeto a:

- **Restricciones de precedencia:** Cada bloque puede ser extraído solo si todos los bloques que le preceden (P_i) han sido extraídos previamente. Siendo j un bloque que precede a i , la restricción sigue la ecuación (12):

$$x_i \leq x_j \quad \forall j \in P_i \quad (12)$$

- **Restricciones binarias** (ecuación (13)) :

$$x_i \in \{0, 1\} \quad \forall i = 1, 2, \dots, n \quad (13)$$

- **Restricción de capacidad:** La cantidad de bloques minados no debe exceder la capacidad operativa disponible (C), la restricción se observa en la ecuación (14):

$$\sum_{i=1}^N x_i \leq C \quad (14)$$

Este problema pertenece a la clase NP-hard, ya que puede reducirse a un problema de mochila con restricciones de precedencia (precedence-constrained knapsack problem [35]), el cual es

conocido por ser NP-hard. Para un yacimiento con N bloques, no existen algoritmos conocidos capaces de resolverlo exactamente en tiempo polinomial, y el tiempo de ejecución para calcular la solución óptima crece exponencialmente, aunque las restricciones de precedencia reducen significativamente el espacio de soluciones factibles.

A diferencia de otras formulaciones que requieren utilizar exactamente la capacidad disponible, este modelo permite operar por debajo de C sin penalizaciones, reflejando la realidad operativa donde factores externos pueden limitar la producción sin incurrir en costos adicionales significativos. Esta formulación se justifica por varias consideraciones prácticas, tales como: variabilidad operativa, flexibilidad económica, gestión de riesgos e integración con otras etapas de la cadena de valor del mineral.

Esta formulación simplificada mantiene las características esenciales del problema de planificación minera mientras proporciona una base sólida para explorar la aplicabilidad de algoritmos cuánticos variacionales en el dominio minero.

REFORMULACIÓN DEL PROBLEMA DE PLANIFICACIÓN MINERA EN QUBO

Como QAOA requiere que QUBO exprese el problema como una minimización, el problema de planificación minera debe adaptarse. El objetivo original, que buscaba maximizar la ganancia total de los bloques minados, como expresa la ecuación (11), debe reformularse invirtiendo el signo de la función objetivo, tal como se expresa en la ecuación (15):

$$\min_x f(x) = -Z = -\sum_{i=1}^N V_i x_i \quad (15)$$

Luego, las restricciones se incorporan a la función objetivo como términos de penalización, de esta manera el problema es resuelto sin restricciones explícitas. Para una restricción de la forma $g(x) \leq 0$, se añade el término $\lambda[g(x)]^2$ a la función objetivo, penalizando si se viola la restricción. Para garantizar la penalización efectiva, un enfoque es utilizar una constante λ lo suficientemente grande para aumentar el valor de la función objetivo de manera significativa cuando se violen las restricciones, haciendo que estas soluciones sean indeseables durante la optimización [36]. El valor de λ debe elegirse cuidadosamente [37], ya que un valor demasiado pequeño no garantiza que se respeten las restricciones en la solución, y un valor demasiado elevado puede causar problemas matemáticos durante la optimización.

La naturaleza binaria de QUBO captura intrínsecamente la restricción de la ecuación (13) sin agregarla a la función objetivo, sin embargo, las restricciones en la formulación del problema de planificación minera, representadas como $x_i \leq x_j$ en la ecuación (12), se incorporan a la función objetivo mediante el término de penalización $\lambda_{i,j}(x_i - x_j)$ [37]. Por ejemplo, considerando la configuración de cuatro bloques: A, B, C, D que se puede ver en la Figura 1, las restricciones de precedencia son: $x_D \leq x_A$, $x_D \leq x_B$, $x_D \leq x_C$:

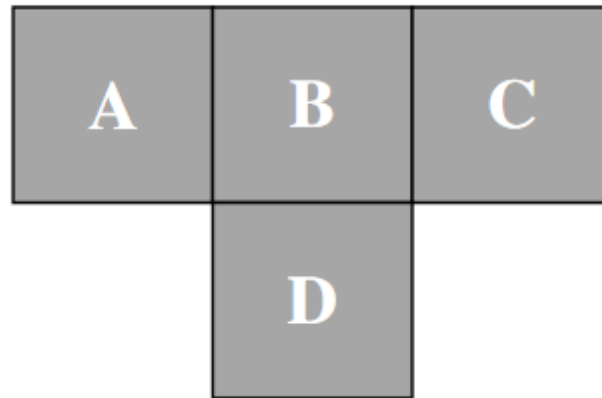


Figura 1. Ejemplo simple en 2D del problema de planificación.

Estas restricciones se traducen en la función objetivo añadiendo el término: $\lambda_{D,A}(x_D - x_D x_A) + \lambda_{D,B}(x_D - x_D x_B) + \lambda_{D,C}(x_D - x_D x_C)$.

De manera general, la restricción de precedencia se puede modelar por la ecuación (16):

$$\sum_{i=1}^N \sum_{j \in P_i} \lambda_{i,j} (1 - x_j) x_i \quad (16)$$

Luego, a la restricción de capacidad se le debe agregar una variable de holgura para representar una restricción de igualdad, como se observa en la ecuación (17):

$$\sum_{i=1}^N x_i + s = C \quad (17)$$

Dado que se requieren variables binarias, la variable de holgura s debe representar el valor para alcanzar C , pero en número binario. Para esto, se utiliza la representación binaria estándar, ecuación (18):

$$s = \sum_{k=0}^{K-1} 2^k s_k, \quad K = \lceil \log_2(C + 1) \rceil \quad (18)$$

Donde:

- $s_k \in \{0, 1\}$ son variables binarias auxiliares,
- K es el número mínimo de bits necesarios para representar todos los valores posibles de s .
- 2^k representa el peso binario de cada bit.

Para dejar esta restricción en la función objetivo, se agrega el término de la ecuación (17), en donde β una constante positiva que penaliza la violación de la restricción. El término de penalización se observa en la ecuación (19).

$$\beta \left(\sum_{i=1}^N x_i + \sum_{k=0}^{K-1} 2^k s_k - C \right)^2 \quad (19)$$

De esta manera, combinando la maximización de la ganancia de la ecuación (15) y los términos de penalización de la ecuación (16) y ecuación (19), se construye la función objetivo (ecuación (20)):

$$\min_{x,s} f(x, s) = -\sum_{i=1}^N V_i x_i + \sum_{i=1}^N \sum_{j \in P_i} \lambda_{i,j} (1 - x_j) x_i + \beta \left(\sum_{i=1}^N x_i + \sum_{k=0}^{K-1} 2^k s_k - C \right)^2 \quad (20)$$

Los elementos de la matriz Q se codifican de la siguiente manera, siguiendo la ecuación (3) y ecuación (20).

- **Elementos diagonales:** Para las variables originales x_i , ($i = 1, \dots, N$) se sigue la ecuación (21):

$$q_{i,i} = -V_i + \sum_{j \in P_i} \lambda_{i,j} + \beta(1 - 2C) \quad (21)$$

Para las variables de holgura s_k , ($k = 0, \dots, K - 1$), se sigue la ecuación (22):

$$q_{N+1+k, N+1+k} = \beta(2^{2k} - 2C \cdot 2^k) \quad (22)$$

- **Elementos fuera de la diagonal:** Entre variables originales x_i , ($i = 1, \dots, N$) y x_j , ($j = 1, \dots, N$) se sigue la ecuación (23):

$$q_{i,j} = q_{j,i} = \frac{2\beta - \lambda_{i,j} \cdot I_{precedencia}(i, j)}{2} \quad (23)$$

Donde $I_{precedencia}(i, j) = 1$ si existe la relación de precedencia entre i y j , 0 en caso contrario. Entre variable original x_i y variable de holgura s_k se utiliza la ecuación (24):

$$q_{i, N+1+k} = q_{N+1+k, i} = \beta \cdot 2^k \quad (24)$$

Entre variables de holgura s_k y s_l , se utiliza la ecuación (25):

$$q_{N+1+k, N+1+l} = q_{N+1+l, N+1+k} = \beta \cdot 2^k \cdot 2^l = \beta \cdot 2^{k+l} \quad (25)$$

Por ejemplo, siguiendo el caso de prueba de la Figura 1, siendo los bloques $V_A = -10$, $V_B = 5$, $V_C = -100$ y $V_D = 300$. Asumiendo $\beta = 1000$; dado que todas las restricciones de precedencia tienen la misma importancia $\lambda_{D,A} = \lambda_{D,B} = \lambda_{D,C} = \lambda = 1000$; y finalmente un límite de la capacidad de

extracción $C=3$, la matriz Q se obtiene siguiendo la siguiente formulación:

Problema original:

- **Formulación original** (ecuación (26)) :

$$\max_x Z = -10x_A + 5x_B - 100x_C + 300x_D \quad (26)$$

Sujeto a las ecuaciones (27)-(31):

$$x_D \leq x_A \quad (\text{precedencia}) \quad (27)$$

$$x_D \leq x_B \quad (\text{precedencia}) \quad (28)$$

$$x_D \leq x_C \quad (\text{precedencia}) \quad (29)$$

$$x_A + x_B + x_C + x_D \leq 3 \quad (\text{capacidad}) \quad (30)$$

$$x_i \in \{0, 1\} \text{ para } i \in \{A, B, C, D\} \quad (31)$$

Paso 1: Transformación de maximización a minimización (ecuación (32)).

$$\min_x f(x) = 10x_A - 5x_B + 100x_C - 300x_D \quad (32)$$

Paso 2: Conversión de restricciones a penalizaciones.

- **Restricciones de Precedencia** (ecuaciones (33)-(35)) :

$$x_D \leq x_A \rightarrow \text{Penalización: } \lambda(x_D - x_D x_A) \quad (33)$$

$$x_D \leq x_B \rightarrow \text{Penalización: } \lambda(x_D - x_D x_B) \quad (34)$$

$$x_D \leq x_C \rightarrow \text{Penalización: } \lambda(x_D - x_D x_C) \quad (35)$$

Total precedencias: $\lambda[3x_D - x_D(x_A + x_B + x_C)]$

- **Restricciones de capacidad:** $x_A + x_B + x_C + x_D \leq 3$

Introducir variables de holgura:

Para $C=3$, necesitamos $K = \lceil \log_2(4) \rceil = 2$ variables binarias: s_0, s_1 , debido a que $s = s_0 + 2s_1$ puede representar valores $\{0, 1, 2, 3\}$

Restricción modificada (ecuación (36)):

$$x_A + x_B + x_C + x_D + s_0 + 2s_1 = 3 \quad (36)$$

Penalización (ecuación (37)):

$$\beta(x_A + x_B + x_C + x_D + s_0 + 2s_1 - 3)^2 \quad (37)$$

Paso 3: Función objetivo QUBO completa (ecuación (38)).

$$\begin{aligned} \min_{x,s} f(x, s) = & 10x_A - 5x_B + 100x_C - 300x_D + 1000(3x_D - x_Dx_A - x_Dx_B - x_Dx_C) \\ & + 1000(x_A + x_B + x_C + x_D + s_0 + 2s_1 - 3)^2 \end{aligned} \quad (38)$$

Paso 4: Expansión y simplificación.

• **Términos lineales (diagonales)** (ecuaciones (39)-(44)):

$$Q_{AA} = 10 + 1000(1 - 6) = -4990 \quad (39)$$

$$Q_{BB} = -5 + 1000(1 - 6) = -5005 \quad (40)$$

$$Q_{CC} = 100 + 1000(1 - 6) = -4900 \quad (41)$$

$$Q_{DD} = -300 + 3000 + 1000(1 - 6) = -2300 \quad (42)$$

$$Q_{s_0,s_0} = 1000(2^{2 \cdot 0} - 6 \cdot 2^0) = -5000 \quad (43)$$

$$Q_{s_1,s_1} = 1000(2^{2 \cdot 1} - 6 \cdot 2^1) = -8000 \quad (44)$$

• **Términos cuadráticos (fuera de la diagonal)** (ecuaciones (45)-(49)):

Entre bloques sin precedencia:

$$\frac{2\beta}{2} = 1000 \quad (45)$$

Entre bloques con precedencia:

$$\frac{2\beta - \lambda}{2} = 500 \quad (46)$$

Entre bloques y s_0 :

$$\frac{2\beta \cdot 1}{2} = 1000 \quad (47)$$

Entre bloques y s_1 :

$$\frac{2\beta \cdot 2}{2} = 2000 \quad (48)$$

Entre s_0 y s_1 :

$$\frac{2\beta \cdot 2}{2} = 2000 \quad (49)$$

Paso 5: Matriz Q final (Figura 2).

• **Variables ordenadas:** $[x_A, x_B, x_C, x_D, s_0, s_1]$

$$Q = \begin{bmatrix} -4990 & 1000 & 1000 & 500 & 1000 & 2000 \\ 1000 & -5005 & 1000 & 500 & 1000 & 2000 \\ 1000 & 1000 & -4900 & 500 & 1000 & 2000 \\ 500 & 500 & 500 & -2300 & 1000 & 2000 \\ 1000 & 1000 & 1000 & 1000 & -5000 & 2000 \\ 2000 & 2000 & 2000 & 2000 & 2000 & -8000 \end{bmatrix}$$

Figura 2. Matriz Q final del caso de prueba 2D.

Paso 6: Creación de H_c (ecuación (50)).

$$H_{cost} = 1152.5I - 255\sigma_A^Z - 247.5\sigma_B^Z - 300\sigma_C^Z - 1100\sigma_D^Z - 500\sigma_{s_0}^Z - 1000\sigma_{s_1}^Z - 250\sigma_A^Z\sigma_D^Z - 250\sigma_B^Z\sigma_D^Z - 250\sigma_C^Z\sigma_D^Z \quad (50)$$

Paso 7: Interpretación de posibles soluciones.

• **Análisis de factibilidad:**

Ejemplo de solución candidata: $x_A = 1, x_B = 1, x_C = 0, x_D = 1, s_0 = 0, s_1 = 0$.

Verificación:

- **Precedencias:** Como $x_D = 1$ pero $x_C = 0$, se viola $x_D \leq x_C$.
- **Capacidad:** $1 + 1 + 0 + 1 + 0 + 2 \cdot 0 = 3 \leq 3$.

Otro ejemplo de solución factible: $x_A = 1, x_B = 1, x_C = 1, x_D = 1, s_0 = 1, s_1 = 0$.

Verificación:

- **Precedencias:** Todas satisfechas.
- **Capacidad:** $1+1+1+1+1+2 \cdot 0=4>3$, restricción violada.

Nota: Las penalizaciones λ y β forzarán soluciones que respeten todas las restricciones.

CASO DE ESTUDIO

Para validar la efectividad del enfoque cuántico propuesto, se implementó una instancia de prueba que examina un yacimiento ficticio compuesto por 18 bloques organizados en una estructura tridimensional. El caso corresponde a un yacimiento de cobre oxidado de baja ley que se busca extraer mediante el método a cielo abierto, el cual será procesado a través de un proceso combinado de Lixiviación (LX) y Extracción por Solventes (SX) / Electroobtención (EW).

El valor de cada bloque se calcula utilizando la ecuación (51):

$$v_b = \begin{cases} m_b \times r \times p - t_b \times (c_{\min} + c_{\text{pro}}), & \text{ley} \geq \text{corte}, \\ -t_b \times c_{\min}, & \text{ley} < \text{corte} \end{cases} \quad (51)$$

Donde:

- m_b : cantidad de metal contenido en el bloque b .
- r : tasa de recuperación de metal del proceso.
- p : precio de mercado del metal.
- t_b : tonelaje total del bloque b .
- $c_{\min}$: costo por tonelaje minado.
- c_{pro} : costo por tonelaje procesado.
- ley : porcentaje de metal contenida en el bloque.
- corte : ley mínima que determina la rentabilidad de un bloque.

La [Tabla 1](#) presenta los valores de los parámetros utilizados. La optimización se implementó en Python, empleando Qiskit sobre un simulador cuántico, el cual se utilizó en lugar de hardware cuántico real debido a las restricciones de acceso. El simulador se ejecutó en un equipo con procesador Intel Core i5-8300H, tarjeta gráfica Nvidia GTX 1050 y 16 GB de memoria RAM.

Tabla 1. Parámetros del caso de estudio.

Parámetro	Valor
Número de bloques	3 x 3 x 2
Dimensión del bloque (m)	5 x 5 x 2,5
Peso del bloque (tons)	150
Precio del metal (USD/lb)	4,32
Recuperación (%)	87
Costo de Minado (USD/ton)	3
Costo de Procesamiento (USD/ton)	5

Tabla 1. Parámetros del caso de estudio. (continuación)

Parámetro	Valor
Ley de Corte (%)	0,09

Formulación QUBO específica

Aplicando la metodología descrita en la sección anterior, el problema se formuló con las siguientes características:

- **Variables binarias:** 18 bloques distribuidos en capas tridimensionales.
- **Bloques factibles:** Las restricciones de precedencia geológica redujeron el espacio de decisión a 10 bloques con potencial de extracción.
- **Parámetros de penalización:** $\lambda=\beta=1000$, seleccionados para que las violaciones de restricciones sean fuertemente prohibitivas.
- **Capacidad de minado:** Es posible extraer 9 bloques.
- **Variables de holgura:** $K = 4$ variables binarias necesarias para representar la capacidad no utilizada.
- **Variables totales:** 14 variables binarias (10 bloques + 4 variables de holgura).
- **Restricciones de precedencia:** Cada bloque en una capa inferior requiere de la extracción previa de hasta 9 bloques precedentes en la capa superior, reflejando la característica tridimensional del depósito.

Debido a la cantidad de bloques factibles mencionados (10 bloques), sumado a las 4 variables de holgura, el caso de prueba tridimensional resultó en una matriz Q de dimensión 14×14 , cuya representación completa es la siguiente (Figura 3):

$$Q = \begin{bmatrix} -17010 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 500 & 1000 & 2000 & 4000 & 8000 \\ 1000 & -17005 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 500 & 1000 & 2000 & 4000 & 8000 \\ 1000 & 1000 & -16900 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 500 & 1000 & 2000 & 4000 & 8000 \\ 1000 & 1000 & 1000 & -17300 & 1000 & 1000 & 1000 & 1000 & 1000 & 500 & 1000 & 2000 & 4000 & 8000 \\ 1000 & 1000 & 1000 & 1000 & -17020 & 1000 & 1000 & 1000 & 1000 & 500 & 1000 & 2000 & 4000 & 8000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & -17050 & 1000 & 1000 & 1000 & 500 & 1000 & 2000 & 4000 & 8000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & -17100 & 1000 & 1000 & 500 & 1000 & 2000 & 4000 & 8000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & -17030 & 1000 & 500 & 1000 & 2000 & 4000 & 8000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & -17090 & 500 & 1000 & 2000 & 4000 & 8000 \\ 500 & 500 & 500 & 500 & 500 & 500 & 500 & 500 & 500 & -8500 & 1000 & 2000 & 4000 & 8000 \\ 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & 1000 & -17000 & 2000 & 4000 & 8000 \\ 2000 & 2000 & 2000 & 2000 & 2000 & 2000 & 2000 & 2000 & 2000 & 2000 & 2000 & -32000 & 8000 & 16000 \\ 4000 & 4000 & 4000 & 4000 & 4000 & 4000 & 4000 & 4000 & 4000 & 4000 & 4000 & 8000 & -56000 & 32000 \\ 8000 & 8000 & 8000 & 8000 & 8000 & 8000 & 8000 & 8000 & 8000 & 8000 & 8000 & 16000 & 32000 & -80000 \end{bmatrix}$$

Figura 3. Matriz Q final del caso de estudio tridimensional.

Implementación QAOA

El Hamiltoniano de costo QAOA se construyó de manera directa a partir de la matriz QUBO, siguiendo la transformación estándar de la ecuación (1).

En el marco de la implementación realizada en Qiskit, se fijaron los siguientes parámetros de la optimización:

- Optimizador clásico: COBYLA (Constrained Optimization BY Linear Approximation) con máximo 50 iteraciones.

- Operador mezclador: Estándar $B = \sum_{i=1}^{14} \sigma_i^x$
- Estado inicial: Superposición uniforme: $|\psi_0\rangle = |+\rangle^{\otimes 14}$

En contraste, los siguientes parámetros fueron explorados de manera experimental:

- Profundidad del circuito: 9, determinado experimentalmente.
- Parámetros variacionales : $2p=18$ ángulos optimizados de manera clásica en cada ejecución.

Las múltiples restricciones de precedencia, junto con las restricciones de capacidad, crearon una matriz Q con alta densidad de términos no nulos que interactúan entre sí. A esto se sumó la incorporación de variables de holgura, las cuales introdujeron una complejidad adicional y ampliaron significativamente el espacio de soluciones factibles, al permitir múltiples configuraciones viables que cumplen con la restricción de capacidad como cota superior.

Como consecuencia directa de esta complejidad, fue necesario escalar progresivamente la profundidad del circuito p mediante experimentación. Se evaluaron distintos valores de $p \in \{4, 9, 14, 21\}$ y se observó que profundidades bajas no lograban converger de manera consistente al óptimo global (con iteraciones que resultaban en un valor subóptimo de USD 600). Finalmente, la mayor consistencia se obtuvo en $p=9$, profundidad a partir de la cual el *ansatz* estándar de QAOA fue capaz de converger al óptimo global consistentemente en esta instancia y bajo simulación.

RESULTADOS

Los resultados de QAOA permitieron alcanzar una solución con un valor de USD 605. Esta solución fue validada de forma independiente utilizando *Gurobi Optimizer* [38] y se confirmó que el algoritmo cuántico logró encontrar la configuración óptima global de bloques a extraer bajo la formulación propuesta. La configuración final se muestra en la [Figura 4](#).

Es importante destacar que el resultado óptimo obtenido es consistente con la ejecución en un simulador clásico, el cual explora el mismo espacio de amplitudes cuánticas con exactitud numérica. En consecuencia, la obtención del óptimo global no implica ventaja computacional respecto de métodos clásicos, sino que refleja la naturaleza moderada del tamaño del problema y la fidelidad inherente al proceso de simulación.

El éxito en encontrar la solución óptima en esta instancia demuestra la capacidad del *ansatz* estándar de QAOA para resolver problemas de optimización combinatoria con restricciones complejas, y converger a la solución exacta valida la formulación QUBO desarrollada. Sin embargo, es importante aclarar que el proceso no estuvo exento de dificultades. Durante la implementación se observaron inconsistencias en los resultados del algoritmo, y se tuvo que escalar la profundidad del circuito de manera experimental. Esto refleja la sensibilidad del algoritmo a la naturaleza del espacio de parámetros variacionales y su correcta optimización.

Para evaluar la robustez del enfoque, se ejecutó el algoritmo treinta veces de manera independiente. Los resultados globales se resumen en la [Tabla 2](#).

Tabla 2. Evidencia experimental.

Métrica	Resultado
Porcentaje de éxito	100%
Gap promedio	0
Gap máximo	0
Tiempo promedio por ejecución	3 min 57.09 seg
Tiempo total	1 hora 59 min
Memoria promedio utilizada	67.6386 MB

Los valores de *gap* con $p=9$ muestran que QAOA no solo encontró soluciones factibles, sino que reprodujo el óptimo global con total precisión. Esta consistencia es especialmente relevante considerando la presencia de variables de holgura, cuyo rol es expandir el espacio factible para representar correctamente restricciones del tipo “menor o igual”.

Resulta importante destacar que la limitación de 14 qubits, se debió tanto a la estructura del problema (dado que las variables de holgura requieren y expanden el uso de qubits), como a las limitaciones computacionales. Durante el desarrollo se encontró que simular problemas con mayor número de qubits resultaba en fallos del kernel de Python, debido a que los requerimientos de memoria asociados al tamaño de la matriz Q y a la profundidad del circuito QAOA junto al tamaño de los vectores γ y β , excedían los recursos disponibles del sistema. Esto desembocó en un desafío para la escalabilidad de estos algoritmos [39], [40].

En conjunto, los resultados obtenidos muestran que, bajo la profundidad del *ansatz* y los parámetros adecuados, el algoritmo QAOA puede resolver de manera consistente y exacta problemas de planificación minera formulados como QUBO, incluso en escenarios con restricciones complejas y capacidad flexible.

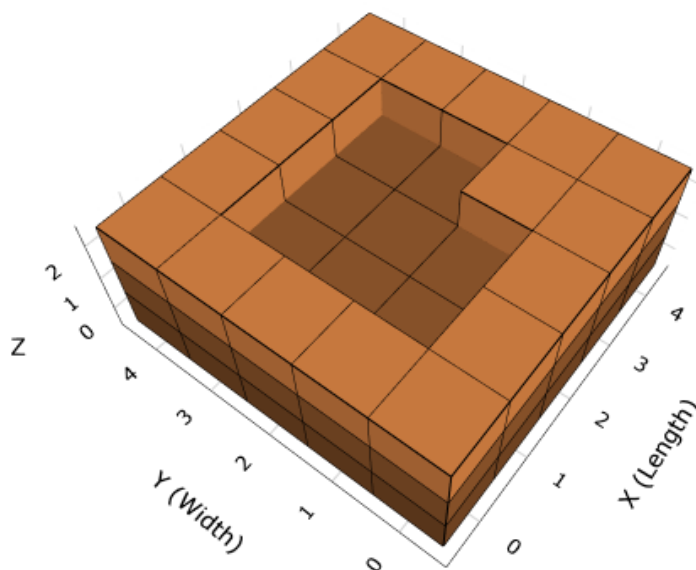


Figura 4. Resultado del caso de estudio.

CONCLUSIONES

Este estudio exploratorio introduce la primera formulación QUBO para planificación minera estratégica con capacidad flexible, resuelta mediante el algoritmo QAOA e implementada sobre un caso de estudio sintético. Los experimentos muestran que la incorporación de variables de holgura permite representar de manera efectiva restricciones operativas del tipo “menor o igual”, manteniendo la calidad de la solución óptima. Asimismo, los resultados obtenidos evidencian que QAOA, bajo una configuración adecuada de parámetros y un *ansatz* lo suficientemente expresivo, es capaz de recuperar de forma consistente la solución exacta del problema, lo que valida tanto el *ansatz* utilizado como la formulación QUBO desarrollada.

En términos cuantitativos, el algoritmo alcanzó un 100% de éxito en treinta ejecuciones independientes. El tiempo promedio de cómputo por ejecución fue de 3 minutos y 57.09 segundos, con un uso promedio de 67.64 MB de memoria RAM, mientras que el tiempo total requerido para todas las ejecuciones fue de 1 hora y 59 minutos. Aunque este trabajo no busca comparar el rendimiento con métodos clásicos, estas métricas permiten dimensionar el costo computacional de la simulación clásica de QAOA y establecen una referencia cuantitativa clara para estudios futuros que incorporen análisis comparativos o mejoras en la eficiencia del modelo.

A partir de estos hallazgos, se identifican varias líneas de desarrollo futuro que pueden abordarse de manera priorizada. En el corto plazo, las mejoras más relevantes corresponden al análisis de otros yacimientos ficticios y la adopción de estrategias más sistemáticas para la selección de penalizaciones [41], [42]. Estos avances no dependen del hardware cuántico y pueden implementarse directamente en simulación clásica.

En el mediano plazo, resulta prometedor evaluar el comportamiento del modelo en hardware cuántico real bajo técnicas de mitigación de ruido. No obstante, para que el caso de estudio utilizado en este trabajo pueda ejecutarse en dispositivos NISQ, será necesario alcanzar hitos tecnológicos específicos, incluyendo fidelidades de puerta de uno y dos qubits cercanas al 99.9%, tiempos de coherencia T_1 y T_2 superiores a 200 μ s, profundidades de circuito operables en el rango $p \leq 10$ -20, y la disponibilidad de aproximadamente 30-50 qubits lógicos efectivos (equivalentes a 100-200 qubits físicos bajo mitigación de errores). Estas capacidades permitirían ejecutar versiones reducidas o adaptadas del modelo en hardware real.

Finalmente, la expansión hacia modelos multiperíodo o formulaciones estocásticas constituye un objetivo de largo plazo, cuya viabilidad dependerá de avances sustantivos en corrección de errores cuánticos, mejoras en compilación cuántica y la disponibilidad de dispositivos con cientos de qubits lógicos. Estas extensiones abrirían la posibilidad de abordar problemas de escala industrial que hoy se encuentran fuera del alcance de las arquitecturas NISQ.

En conjunto, los resultados de este trabajo delinean una ruta de avance para el uso de algoritmos cuánticos variacionales en planificación minera estratégica. Aunque la realización plena de este enfoque depende de progresos futuros en hardware y teoría cuántica, el presente estudio establece una base sólida sobre la cual construir soluciones híbridas cuántico-clásicas que, en el futuro, podrían contribuir de manera significativa a la optimización de proyectos mineros.

Financiamiento

Los autores declaran no haber recibido financiamiento para este trabajo.

REFERENCIAS

- [1] M. Kumral, "Production planning of mines: Optimisation of block sequencing and destination," *International Journal of Mining, Reclamation and Environment*, vol. 26, no. 2, pp. 93–103, 2012, <https://doi.org/10.1080/17480930.2011.644474> [↑]
- [2] A. Newman, E. Rubio, R. Caro, A. Weintraub, and K. Eurek, "A review of operations research in mine planning," *Interfaces*, vol. 40, no. 3, pp. 222–245, 2010, <https://doi.org/10.1287/inte.1090.0492> [↑]
- [3] L. Adler, *SME Mining Engineering Handbook*, 3rd ed. P. Darling, Ed., Englewood, CO, USA: Society for Mining, Metallurgy, and Exploration, 2011. [↑]
- [4] K. Dagdelen and T. B. Johnson, "Optimum open pit mine production scheduling by Lagrangian parameterization," in *Proceedings of the 19th Symposium on Application of Computers and Operations Research in the Minerals Industry (APCOM)*, 1986, pp. 127–142. [↑]
- [5] D. Bienstock and M. Zuckerberg, "Solving LP relaxations of large-scale precedence constrained problems," in *Integer Programming and Combinatorial Optimization*, F. Eisenbrand and F. B. Shepherd, Eds., Berlin, Germany: Springer, 2010, pp. 1–14, https://doi.org/10.1007/978-3-642-13036-6_1 [↑]
- [6] T. Lagos, M. Armstrong, T. Homem-de-Mello, G. Lagos, and D. Sauré, "A framework for adaptive open-pit mining planning under geological uncertainty," *Optimization and Engineering*, vol. 23, no. 1, pp. 111–146, 2022, <https://doi.org/10.1007/s11081-020-09557-0> [↑]
- [7] A. Quelopana, J. Órdenes, R. Araya, and A. Navarra, "Geometallurgical detailing of plant operation within open-pit strategic mine planning," *Processes*, vol. 11, no. 2, Art. no. 381, 2023, <https://doi.org/10.3390/pr11020381> [↑]
- [8] A. Khan, "Long-term production scheduling of open pit mines using particle swarm and bat algorithms under grade uncertainty," *Journal of the Southern African Institute of Mining and Metallurgy*, vol. 118, no. 4, 2018, <https://doi.org/10.17159/2411-9717/2018/v118n4a5> [↑]
- [9] Z. Levinson and R. Dimitrakopoulos, "Simultaneous stochastic optimisation of an open-pit gold mining complex with waste management," *International Journal of Mining, Reclamation and Environment*, vol. 34, no. 6, pp. 415–429, 2020, <https://doi.org/10.1080/17480930.2019.1621441> [↑]
- [10] R. Senécal and R. Dimitrakopoulos, "Long-term mine production scheduling with multiple processing destinations under mineral supply uncertainty, based on multi-neighbourhood Tabu search," *International Journal of Mining, Reclamation and Environment*, vol. 34, no. 7, pp. 459–475, 2020, <https://doi.org/10.1080/17480930.2019.1595902> [↑]
- [11] A. Quelopana Retamal and A. Navarra, "Incorporating operational modes into long-term open-pit mine planning under geological uncertainty: An optimization combining variable neighborhood descent with linear programming," *Mining, Metallurgy & Exploration*, vol. 41, no. 5, 2024, <https://doi.org/10.1007/s42461-024-01052-9> [↑]
- [12] J. L. V. Mariz, M. M. Badiozamani, R. de L. Peroni, and R. M. de A. Silva, "A critical review of bench aggregation and mining cut clustering techniques based on optimization and artificial intelligence to enhance the open-pit mine planning," *Engineering Applications of Artificial Intelligence*, vol. 133, Art. no. 108334, 2024, <https://doi.org/10.1016/j.engappai.2024.108334> [↑]

- [13] R. Noriega and Y. Pourrahimian, “A systematic review of artificial intelligence and data-driven approaches in strategic open-pit mine planning,” *Resources Policy*, vol. 77, Art. no. 102727, 2022, <https://doi.org/10.1016/j.resourpol.2022.102727> [†]
- [14] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Harlow, UK: Pearson, 2021. [†]
- [15] T. Agrawal, *Hyperparameter Optimization in Machine Learning: Make Your Machine Learning and Deep Learning Models More Efficient*, Berkeley, CA, USA: Apress, 2021, <https://doi.org/10.1007/978-1-4842-6579-6> [†]
- [16] R. Campos, “Hybrid quantum-classical algorithms,” 2024, arXiv: 2406.12371. [†]
- [17] K. Blekos, D. Brand, A. Ceschini, C. H. Chou, R.H. Li, K. Pandya, and A. Summer, “A review on Quantum Approximate Optimization Algorithm and its variants,” *Physics Reports*, vol. 1068, pp. 1–66, 2024, <https://doi.org/10.1016/j.physrep.2024.03.002> [†]
- [18] G. Benenti, G. Casati, D. Rossini, and G. Strini, *Principles of Quantum Computation and Information*, Singapore: World Scientific, 2018, <https://doi.org/10.1142/10909> [†]
- [19] F. Glover et al., *The Quadratic Unconstrained Binary Optimization Problem. Theory, Algorithms, and Applications*, A. P. Punnen, Ed., Cham, Switzerland: Springer, 2022, <https://doi.org/10.1007/978-3-031-04520-2> [†]
- [20] M. Hess, L. Palackal, A. Awasthi, and K. Wintersperger, “Effective embedding of integer linear inequalities for variational quantum algorithms,” in *2024 IEEE International Conference on Quantum Computing and Engineering (QCE)*, Montreal, QC, Canada, 2024, pp. 221–231, <https://doi.org/10.1109/QCE60285.2024.00035> [†]
- [21] A. Javadi-Abhari et al., “Quantum computing with Qiskit,” 2024, arXiv:2405.08810. [†]
- [22] T. Hao, X. Huang, C. Jia, and C. Peng, “A quantum-inspired tensor network algorithm for constrained combinatorial optimization problems,” *Frontiers in Physics*, vol. 10, Art. no. 906590, 2022, <https://doi.org/10.3389/fphy.2022.906590> [†]
- [23] Y. Hindy, J. Pointing, M. Tolunay, S. Venkatarao, M. Motta, and J. A. Latone, “A quantum computational approach to the open-pit mining problem,” 2021, arXiv:2107.11345. [†]
- [24] G. Paradezhenko, A. Pervishko, and D. Yudin, “Quantum-assisted open-pit optimization,” *JETP Letters*, vol. 119, no. 6, pp. 470–478, 2024, <https://doi.org/10.1134/S0021364023604256>. [†]
- [25] M. A. Nielsen, I. L. Chuang, and L. K. Grover, “Quantum computation and quantum information,” *American Journal of Physics*, vol. 70, no. 5, pp. 558–559, 2002, <https://doi.org/10.1119/1.1463744> [†]
- [26] T. Albash and D. A. Lidar, “Adiabatic quantum computation,” *Reviews of Modern Physics*, vol. 90, no. 1, Art. no. 015002, 2018, <https://doi.org/10.1103/RevModPhys.90.015002> [†]
- [27] V. Lahtinen and J. K. Pachos, “A short introduction to topological quantum computation,” *SciPost Physics*, vol. 3, no. 3, 2017, <https://doi.org/10.21468/SciPostPhys.3.3.021> [†]
- [28] S. Buck, R. Coleman, and H. Sargsyan, “Continuous variable quantum algorithms: an introduction,” 2021, arXiv:2107.02151. [†]

- [29] D. Castelvechi, “Underdog technologies gain ground in quantum-computing race,” *Nature*, vol. 614, pp. 400–401, 2023, <https://doi.org/10.1038/d41586-023-00278-9> [†]
- [30] A. D. Corcoles et al., “Challenges and opportunities of near-term quantum computing systems,” *Proceedings of the IEEE*, vol. 108, no. 8, pp. 1338–1352, 2020, <https://doi.org/10.1109/JPROC.2019.2954005> [†]
- [31] S. N. Filippov, S. Maniscalco, and G. García-Pérez, “Scalability of quantum error mitigation techniques: from utility to advantage,” 2024, arXiv:2403.13542. [†]
- [32] F. Albarrán, P. Assmann, A. Delgado, D. Goyeneche, C. Hermann Avigliano, S. Torres, C. Torres y S. Walborn, *Recomendaciones y desafíos para el fortalecimiento del ecosistema cuántico en Chile*, Santiago, Chile: Tecnologías Cuánticas, Ministerio de Ciencia, Tecnología, Conocimiento e Innovación, 2024. [En línea]. Disponible: <https://minciencia.gob.cl/areas/comision-asesora-tecnologias-cuanticas/> [†]
- [33] A. D’Urbano, M. Angelelli, and C. Catalano, “The significance of classical simulations in the adoption of quantum technologies for software development,” in *Proceedings of the 24th International Conference on Product-Focused Software Process Improvement (PROFES)*, Dornbirn, Austria, Dec. 10–13, 2023, pp. 60–67, https://doi.org/10.1007/978-3-031-49269-3_6 [†]
- [34] A. Lucas, “Ising formulations of many NP problems,” *Frontiers in Physics*, vol. 2, 2014, <https://doi.org/10.3389/fphy.2014.00005> [†]
- [35] N. Samphaiboon and Y. Yamada, “Heuristic and exact algorithms for the precedence-constrained knapsack problem,” *Journal of Optimization Theory and Applications*, vol. 105, no. 3, pp. 659–676, 2000, <https://doi.org/10.1023/A:1004649425222> [†]
- [36] I. Griva, S. G. Nash, and A. Sofer, *Linear and Nonlinear Optimization*, 2nd ed. Philadelphia, PA, USA: SIAM, 2009. [†]
- [37] F. Glover, G. Kochenberger, R. Hennig, and Y. Du, “Quantum bridge analytics I: A tutorial on formulating and using QUBO models,” *Annals of Operations Research*, vol. 314, no. 1, pp. 141–183, Jul. 2022, <https://doi.org/10.1007/s10479-022-04634-2> [†]
- [38] Gurobi Optimization, “Gurobi Optimizer,” [gurobi.com](https://www.gurobi.com/). Accessed: Oct. 2025. [Online]. Available: <https://www.gurobi.com/> [†]
- [39] E. Farhi, J. Goldstone, and S. Gutmann, “A quantum approximate optimization algorithm,” 2014, arXiv: 1411.4028. [†]
- [40] L. Zhou, S.-T. Wang, S. Choi, H. Pichler, and M. D. Lukin, “Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices,” *Physical Review X*, vol. 10, no. 2, Art. no. 021067, 2020, <https://doi.org/10.1103/PhysRevX.10.021067> [†]
- [41] M. Guignard, “Lagrangian relaxation,” *Top*, vol. 11, no. 2, pp. 151–200, 2003, <https://doi.org/10.1007/BF02579036> [†]
- [42] R. Garnett, *Bayesian Optimization*, Cambridge, MA, USA: Cambridge University Press, 2023. [†]