Technical Report

Niall Carbery, Maya Kamboj

University College Dublin

Abstract

We conduct a comprehensive study on optimizing ship scheduling at the Panama Canal under conditions of freshwater scarcity. To mitigate operational challenges during droughts, we propose a timetable that integrates water-saving measures—such as cross-filling and tandem lockages—by formulating the scheduling problem as a Quadratic Unconstrained Binary Optimization (QUBO). Our approach leverages classical simulated annealing as a proof-of-concept to demonstrate significant reductions in water usage, while also exploring the feasibility of quantum annealing through a neutral atom quantum processor. Furthermore, we incorporate reinforcement learning techniques to dynamically tune QUBO parameters, ensuring an effective balance between operational feasibility and water conservation. The experimental results indicate that our integrated strategy not only enhances water efficiency but also scales favourably with increasing problem size, offering a promising avenue for future quantum-enhanced optimization in critical infrastructure management.

Keywords—QUBO, Quantum Annealing, Freshwater Conservation, Panama Canal

Contents

1 Team Introduction 1
1.1 Niall Carbery . . . 1
1.2 Maya Kamboj . . . 1
2 Introduction 2
2.1 Panama Canal . . . 2
Water-Saving Measures
2.2 SDG Alignment: Sustainable Industries and Transport . . . 3
3 Theory 3
3.1 QUBO . . . 3
Connection to Ising Models
3.2 Quantum Annealing . . . 3
3.3 Neutral Atom Quantum Computers . . . 3
4 Mathematical Formulation 4
4.1 QUBO Binary Variables . . . 4
4.2 Linear Terms . . . 4
4.3 Ship Scheduling Constraint . . . 4
4.4 Tandem Lockage and Combined Length Penalty . . . 4
4.5 Panamax Cross-fill Reward . . . 5
4.6 Choice of penalty terms . . . 5
5 AI Contribution and Methodology 5
5.1 Need for AI Contribution . . . 5
5.2 AI Implementation . . . 5
6 Quantum Computing Integration & Feasibility 5
6.1 Quantum Adiabatic Algorithm (QAA) . . . 5
6.2 Main Algorithmic Steps . . . 6
6.3 Parameters and Their Calculation . . . 6
6.4 Rationale Behind Pulse and Channel Selection . . . 6
6.5 Feasibility and Scalability . . . 6
Removing Detuning Requirement
7 Method 7
7.1 Simulated Implementation . . . 7
Process • Simulation Details
7.2 Quantum Adiabatic Algorithm (QAA) Implementation . . . 7
Design of the Adiabatic Pulse • Simulation of the QAA Evolution • Analysis of Time Evolution Effects
8 Results 8
8.1 Simulated Results . . . 8
8.2 QAA Results . . . 8
8.3 Embedding . . . 8
8.4 QAA . . . 9
8.5 Roadmap and Future Work . . . 9
8.6 Errors in QAA on Nuetral atoms . . . 10
9 Conclusion 10
10 Appendix 10
10.1 GitHub . . . 10
10.2 References . . . 10

1. Team Introduction

1.1. Niall Carbery

As a third-year student at University College Dublin (UCD) pursuing a Bachelor of Science in Theoretical Physics, I have developed a deep interest in quantum science and technology, particularly in quantum computing and its applications in machine learning and optimization. My curiosity was initially sparked by taking modules offered through the university and then applying these methods to algorithm development and optimization, such as designing quantum kernel-based classifiers and estimating ground-state Hamiltonians in quantum computing projects. Looking forward, I hope to further enhance my expertise fields and even consider future study in this field.

1.2. Maya Kamboj

As a third year Theoretical Physics BSc student at University College Dublin (UCD), my exposure to quantum computing began with an academic module covering both theoretical foundations and practical implementations. Through this, I adopted an appreciation for the power of quantum technology and have a budding interest in quantum computing and its applications.

Beyond this, I am particularly interested in the world of mathematical modelling and optimisation problems, having developed my

knowledge in the area by participating in the COMAP Mathematical Contest in Modelling. Looking ahead, I am excited to combine these two areas and explore how quantum algorithms and hybrid quantum-classical approaches can optimize complex systems in the real world, particularly in areas such as engineering.

2. Introduction

2.1. Panama Canal

The Panama Canal is a pivotal maritime route that connects the Atlantic and Pacific Oceans, significantly reducing travel distances for global shipping. By enabling vessels to bypass the lengthy and treacherous journey around the southern tip of South America, the canal has been instrumental in facilitating international trade and bolstering economic growth since its completion in 1914.[1]. The canal saves roughly 13,000 kilometres on the journey, reducing CO_2 emissions for the journey.

In recent years, the Panama Canal has faced significant challenges due to water scarcity, from the more frequent droughts. These conditions have led to decreased water levels in Gatún Lake, threatening both the canal's operational capacity and the freshwater supply for local populations[2]. This has led to ships making the longer journey around South America, releasing more CO_2. The aim of our project is to implement a timetable that incorporates water saving measures to increase the amount of crossings possible during drought periods, therefore reducing the amount of CO_2 released in global trade.

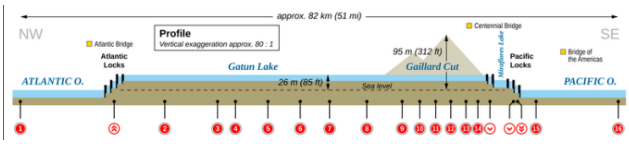

Figure 1. Profile of the Panama Canal *(Image by Thoroe, 2012, via Wikimedia Commons, "Map of the Panama Canal (English version)")*

The canal operates using a system of locks that function as water elevators, lifting ships from sea level to the elevation of Gatún Lake (approximately 26 meters above sea level)[1] and then lowering them back to sea level on the opposite side. Each transit through the locks requires a substantial amount of freshwater, which is sourced from Gatún Lake. This freshwater is essential not only for the canal's operations but also as a primary source of drinking water for the surrounding regions.

Specification wise the canal consists of two main locks, one on the Atlantic side and one on the Pacific side, each capable of handling vessels up to 39.5 meters wide, 294 meters long, and 12 meters deep. In contrast, the Neopanamax locks, constructed as part of the expanded canal system completed in 2016, accommodate much larger ships. These locks are designed to handle vessels up to 49 meters in width, 366 meters in length, and 15.2 meters in depth, allowing for increased cargo capacity and larger container ships.

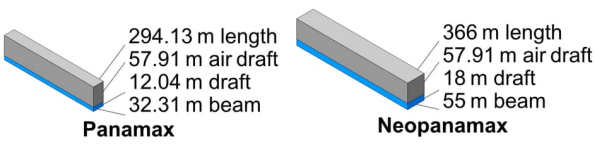

Figure 2. Panama Canal Lock Standards *(Image by Wikimedia Commons contributors, 2016, via Wikimedia Commons "Ship measurements comparison")*

2.1.1. Water-Saving Measures

To save water on a per-crossing basis, there are a number of water saving measures that can be implemented. These are outlined below, but each of them require a large amount of coordination and time tabling. Currently, these water saving measures are not implemented and ships pass through the locks one at a time. This method is simple and does not require much coordination, but implementing them during droughts will increase the number of transits possible given a timetable accounting for them.

Cross-Filling: The original Panamax locks consist of two parallel lanes, each with its own set of chambers. Typically, these lanes are filled independently, drawing water from Gatún Lake and releasing it into the ocean. To conserve water, operators have introduced a cross-filling method, which involves transferring water between the two lanes. This process uses subterranean culverts that have always existed but were only recently repurposed for water conservation.

Figure 3. Cross filling the Miraflores locks, water from the chamber on the right has been released into the chamber on the left without releasing. *(Image by Scope of Work, 2025, via Scope of Work, "Panamaximization")*

Cross-filling is a complex procedure, requiring precise coordination and scheduling to ensure all vessels are positioned correctly. The operations team works closely with ACP pilots and client boat captains to streamline the process. While cross-filling results in some delays compared to standard operations, it saves approximately six lockages worth of water each day, this is against the 32 average crossings made on the canal, or the equivalence of 1.2 billion litres of freshwater.

Tandem Lockages: When vessel sizes permit, two ships transit simultaneously within a single lock chamber. This approach maximizes water usage efficiency by reducing the number of lockages required, thereby conserving water. Details regarding ship lengths must be recorded and coordinated with the available types of locks

Water-Saving Basins: The Neopanamax locks, significantly larger than the original Panamax locks with 2.4 times the capacity, were designed with water conservation in mind. Each chamber is paired with three water-saving basins that recycle water by draining it into the basins and reusing it to fill the next chamber before being released to the sea. Despite their greater volume, these locks use 7%[3] less water per transit compared to the Panamax locks,

However, the water-saving basins cannot always be used, as they introduce a significant issue: saltwater intrusion into Gatún Lake. Saltwater, which naturally enters the canal system through ship movements in the locks, becomes more concentrated in the lake when the basins are used. This occurs because the basins recycle water from the bottom layer of the locks, where denser saltwater settles. Over time, this saltwater seeps upward into Gatún Lake, which serves as both a freshwater reservoir for the canal and a critical drinking water source for Panamanians.

Multimodal Crossings During drought's draft restrictions are put in place, a vessel's draft is the distance from its lowest point to the waterline. Typically, vessels with drafts up to 15.2 meters can transit the canal, in drought times this can be reduced to 13.6 meters. This poses a challenge for ships carrying thousands of containers, but

reducing a vessel's load decreases its draft. To comply, some freight companies are partially unloading containers at ports on either side of the canal and transporting them across the isthmus by rail. The lighter ships can then traverse the lake within the draft restrictions. Some companies, like Maersk, have taken this further by avoiding canal crossings altogether for certain vessels. They unload cargo on one side of the isthmus, transport it by rail, and reload it onto different ships on the other side—creating a multimodal logistics solution.

Prioritisation Prioritising ships that carry perishable goods is a sustainable approach that enhances supply chain efficiency while reducing waste. Perishable cargo—such as fresh produce, dairy, and pharmaceuticals—demands swift delivery to maintain quality and prevent spoilage. By ensuring that these shipments receive preferential treatment in scheduling and transit operations, logistics systems can minimize delays and the energy-intensive measures (such as prolonged refrigeration) required to preserve these goods. This targeted prioritization not only bolsters the economic viability of perishable shipments by reducing loss and spoilage but also contributes to a lower carbon footprint by streamlining operations and conserving resources, thereby supporting broader sustainability goals.

2.2. SDG Alignment: Sustainable Industries and Transport

The sustainable industries and transport challenge focuses on the eco-friendly innovation of transportation and industries to foster a sustainable economy. The core aims of this challenge are adopted from the United Nations Sustainable Development Goals (SDG) and specifically focus on SDG 9 and SDG 12. The goals respectively aim to "Build resilient infrastructure, promote inclusive and sustainable industrialization and foster innovation", and "Ensure sustainable consumption and production patterns". The optimization of ship scheduling at the Panama Canal tackles this SDG challenge by reducing vital resources needed for the operation of the canal. This allows for a more sustainable transport industry within the vital global trade[4]

3. Theory

As before, a timetable needs to be generated that includes water saving measures to increase the amount of crossings possible during drought periods. In order to implement a time tabling service, we create a problem space using a mathematical framework (Quadratic unconstrained binary optimization), the goal is to use a process called quantum annealing, a process that allows for combinatorial optimization problems to be solved on a quantum computer. The need for this development is based on the dynamics and the large possible number of implementations. There is evidence that such models will run exponentially faster than alternative methods.[5–7] In order to solve to find a solution using a quantum computer, the problem space must be mapped Ising problem or sum of spin variables, which can be solved on a quantum computer.

3.1. QUBO

Quadratic unconstrained binary optimization is a combinatorial optimization problem where a vector of binary decision variables is constructed using $f_Q(x)$ below. The QUBO problem relates to finding a binary vector x^* that is minimal with respect to $f_Q(x)$.

$$f_Q(x) = x^T Q x = \sum_{i=1}^{n} \sum_{j=i}^{n} Q_{ij} x_i x_j \quad \forall x \in \mathbb{B}^n \tag{1}$$

Q matrix represents the quadratic coefficients of the objective function, where it relates the interactions between binary variables in the optimization problem. QUBO problems are typically NP-hard.[8] Methods such as linear integer programming or other methods try to find optimal solutions, but complexity grows exponentially in n, as the number of possible binary vectors to be evaluated as $|\mathbb{B}^n| = 2^n$. This exponential growth in complexity necessitates the problems should not be solved using traditional computational methods.

If we encode the binary variables into qubit states on a quantum computer and transverse Hilbert space which is mapped from the search space. The implication being that the quantum computer could solve a NP - complete problem in polynomial time, while this is not proven, the consensus is that it is unlikely. Current research[9] that can approximate solutions in sorter time compared to classical computers in some cases. This is because a quantum computer can exploit shortcuts not available to traditional optimizers such as simulated annealing[10] due to the highly entangled nature of samples that are potentially not storable by classical computers.

3.1.1. Connection to Ising Models

QUBO problems can be converted to Ising problems, this by mapping the binary variables $x_i \in 0, 1$ in QUBO to the spin variables $s_i \in -1, +1$ in the Ising model.

$$H = \sum_{i,i'} J_{i,i'} s_i s_{i'} + \sum_i h_i s_i \tag{2}$$

By substituting $\sigma = 2x - 1$ into the Ising model Hamiltonian, the spin variables are rewritten in terms of binary variables using $x_i^2 = x_i$ and constants do not change the position of the optimal solution.[11] The benefit of such transformation allows for the application of the adiabatic theory.

3.2. Quantum Annealing

Quantum annealers implement the quantum version of the Ising model, where the spin variables are replaced by Pauli matrices acting on the spin's subspace. The Ising representation is the native input for the quantum annealer. The original problem from Eq. (2) becomes a search for the lowest-energy (or near-low-energy, in practice) eigenstate of the following Hamiltonian:

$$H = \sum_{i,i'} J_{i,i'} \sigma_z^i \sigma_z^{i'} + \sum_i h_i \sigma_z^i \tag{3}$$

The core principle of quantum annealing relies on the adiabatic theorem. Initially, the quantum system is prepared in the ground state of an initial simple Hamiltonian $H_0 = \sum_i \sigma_x^i h_i$, then the Hamiltonian evolves slowly to the final Hamiltonian H_p which represents our problem. If this process is slow enough, the system ideally reaches the ground state or a low-energy excited state of the final Hamiltonian, and thus a low-energy state of the Ising model from Eq. (3). The efficiency of this process depends on various factors, such as the eigenvalue distribution and the gap between eigenvalues of the Hamiltonian in Eq. (3). These relationships are complex, and the results can vary when comparing two QUBOs with different penalty coefficients and spectra. However, research strongly suggests that is method is both a more accurate and faster solution finder compared to classical simulated annealing.[5]

3.3. Neutral Atom Quantum Computers

In analogue quantum processors, each atom is approximated as a simple system with two electronic states. The qubits with bases states $|0\rangle$ and $|1\rangle$ being a low energy ground state and highly excited Rydberg state.[12] The evolution of the qubits state is parameterized by time dependent control-fields, $\Omega(t)$ and $\delta(t)$ i.e. the properties of the laser acting on the atoms. The quantum state of atom i can significantly alter the state of atom j, depending on their pair distance r_{ij}. The excitation of i to a Rydberg state $|1\rangle$ shifts the energies of the corresponding Rydberg states of nearby j by an amount $U(r_{ij})$. The latter quantity can be considered impactful compared to the action of the control fields with $r_{ij} \leq r_b$ with r_b the blockade radius. The blockade radius is used as the entangling process in the neutral atom device.[13]

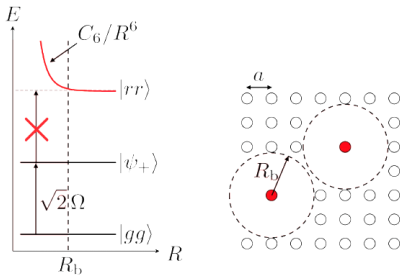

Figure 4. Neutral Atom Rydberg Radius Implementation *(Image by Pasqal)*

Neutral atom QPU's can implement the Ising Hamiltonian with the time dependent control part and position dependent interaction. Here a laser pulse sequence on an entire array N of atoms, located at position $\vec{r}$ have the time evolution described by the Ising Hamiltonian

$$\hat{H} = \hat{H}_{ctrl} + \hat{H}_{int}(\vec{r}) \tag{4}$$

$$= \hbar \sum_{i=1}^{N} \left(\frac{\Omega(t)}{2} \hat{\sigma}_i^x - \delta(t) \hat{n}_i \right) + \frac{1}{2} \sum_{i \neq j} U_{ij} \hat{n}_i \hat{n}_j \tag{5}$$

Where $\hat{\sigma}_i^{\alpha}$ are the Pauli matrices applied on the ith qubit, $\hat{n}_i = \frac{1+\hat{\sigma}_{z_i}}{2}$ is the number of Rydberg excitations (with eigenvalues 0 or 1) on site i. $U_{ij} = \frac{C_6}{r_{ij}^6}$ is the van der Waals interaction, C_6 is the van der Waals coefficient and r_{ij} is the distance between atoms i and j, represents the distance-dependent interaction between qubits i and j. The state of the system is initialized to $|000\ldots0\rangle$. Once the pulse sequence drives the system towards its final state $|\psi\rangle = \sum_{w \in B^N} a_w |w\rangle$, a global measurement is performed through fluorescence imaging: the system is projected to a basis state $|w\rangle$ with probability $|a_w|^2$. The obtained picture reveals which atoms were measured in $|0\rangle$ (bright spot) and which were in $|1\rangle$ (dark spot). Repeating the cycle (loading atoms, applying a pulse sequence, and measuring the register) multiple times constructs a probability distribution that approximates $|a_w|^2$ for $w \in B^N$, allowing to get an estimator of $|\psi\rangle$.

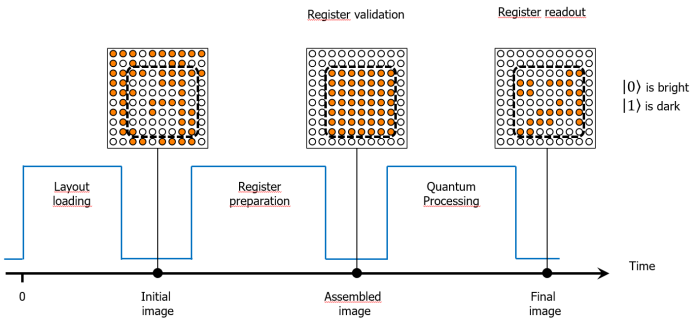

Figure 5. Neutral Atom Quantum Computer *(Image by Pasqal)*

We aim to create a quantum state where the probability amplitudes concentrate on the bit strings that represent the most water-efficient configurations. When we sample from such a state, we effectively steer the optimization process toward solutions with the lowest water costs.

This quantum state can be highly entangled to capture the complex interplay between different water usage factors, and such a state is inherently difficult to simulate or store using classical methods. Neutral atom quantum computers, however, provide remarkable scalability—capable of managing hundreds of atoms[14] at a time—along with a global addressing scheme in analogue mode that makes it straightforward to entangle numerous qubits. In contrast, achieving comparable levels of complexity with quantum circuit-based approaches would require a prohibitive number of quantum gates.

4. Mathematical Formulation

4.1. QUBO Binary Variables

In this work we consider a scheduling problem with N ships and T time slots. To model the timetable, we define the binary decision variable

$$x_{i,t} = \begin{cases} 1, & \text{if ship } i \text{ is scheduled in time slot } t, \\ 0, & \text{otherwise,} \end{cases} \tag{6}$$

for $i = 0, \ldots, N-1$ and $t = 0, \ldots, T-1$. Each time slot is associated with a lock type which determines both the lock capacity and the baseline water cost incurred during transit. Denote by

$$K_t = \text{get\_lock\_length(lock\_types}[t]\text{)}$$

the capacity and by

$$W_t = \text{water\_cost\_for\_slot(lock\_types}[t]\text{)}$$

the water cost for slot t. Furthermore, each ship i is characterized by a benefit B_i and a length L_i. The incompatibility between a ship's physical dimension and the lock capacity is accounted for via an indicator penalty:

$$\mathbb{I}\{L_i > K_t\} P,$$

where P denotes the infeasibility penalty.

The overall objective is cast as a Hamiltonian that simultaneously rewards transits (via the benefit B_i) and penalizes water usage and operational infeasibilities. In particular, the QUBO form of our Hamiltonian is given by the sum of four components:

$$H(x) = Q_{\text{linear}} + Q_{\text{constraint}} + Q_{\text{tandem}} + Q_{\text{crossfill}}. \tag{7}$$

4.2. Linear Terms

The linear term aggregates the benefit of each ship assignment, the water cost per slot, and any penalty incurred when a ship's length exceeds the lock's capacity. It is defined as Q_{linear}

$$\sum_{i=0}^{N-1} \sum_{t=0}^{T-1} \left[-\lambda_{\text{benefit}} B_i + \lambda_{\text{water}} W_t + \mathbb{I}\{L_i > K_t\} P \right] x_{i,t}. \tag{8}$$

Because the objective is minimization, a higher benefit (i.e., larger B_i) reduces the overall energy, while water usage and incompatibility increment the cost.

4.3. Ship Scheduling Constraint

To ensure that each ship is scheduled at most once, we incorporate a quadratic penalty that enforces the constraint:

$$\lambda_{\text{ship}} \left(\sum_{t=0}^{T-1} x_{i,t} - 1 \right)^2 \tag{9}$$

If for any ship the sum over time slots is more than 1, the penalty grows quadratically with the deviation. The penalty term $2\lambda_{ship} x_{i,t}$ is chosen high enough to force feasibility. The linear term when the quadratic is expanded out also contributes to rewarding ship assignment by factor $-\lambda_{ship}$. This is from the expansion of the square for a linear and quadratic terms.

4.4. Tandem Lockage and Combined Length Penalty

For two distinct ships i and j scheduled in the same time slot t, a tandem reward is applied, while also ensuring that their combined lengths do not exceed the slot capacity. This term is modelled as

$$Q_{\text{tandem}} = \sum_{t=0}^{T-1} \sum_{0 \le i < j \le N-1} \left[-\lambda_{\text{tandem}} + \mathbb{I}\{L_i + L_j > K_t\} P \right] x_{i,t}\, x_{j,t}. \tag{10}$$

Here, the incentive $-\lambda_{\text{tandem}}$ encourages the assignment of ships in tandem within the same slot, whereas the penalty P is added if their cumulative length is infeasible relative to the lock capacity.

4.5. Panamax Cross-fill Reward

In order to promote efficient utilization across consecutive time slots—specifically when slots feature complementary lock types (e.g., {Panamax_A, Panamax_B})—a cross-fill reward is introduced. For consecutive slots t and $t+1$ meeting this criterion, the following term is included:

$$Q_{\text{crossfill}} = \sum_{\substack{t=0 \\ \text{Condition}}}^{T-2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (-\lambda_{\text{crossfill}})\, x_{i,t}\, x_{j,t+1} \tag{11}$$

With the Condition that cumulative locks have to both be Panamax Locks.This term effectively incentivizes the pairing of ships across adjacent time slots with the designated lock type configuration by reducing the overall energy when such pairings occur. While possible to squeeze multiple ships into one lock, the frequency of large ships passing through the Panama Canal makes it unlikely, similarly there is a high time cost in setting up multiple ships into one lock and therefore only two ships are used for tandem lockages.

4.6. Choice of penalty terms

Determining suitable values for the penalty parameters, λ_{ship}, $\lambda_{conflict}$, λ_{water} and reward term B_i is a complex challenge, prompting the development of specialized algorithms in recent years[15]. However, these algorithms add computational complexity, which must be accounted for when estimating the total computation time (for instance, identifying the correct quantum encoding and circuits is itself NP-hard[16]). Typically, these methods establish a lower bound for the penalty coefficients needed to enforce constraints by differentiating between feasible and non-feasible regions of the spectrum. Ideally, the penalty for constraint violations should be greater than any potential changes in the objective function. However, in practice, better results are often observed with penalty coefficients smaller than these theoretical lower limits[17]. In such cases, the spectra of feasible and non-feasible solutions overlap, which may enhance the solution process by, for example, smoothing local minima during quantum evolution.

5. AI Contribution and Methodology

5.1. Need for AI Contribution

The integrated QUBO formulation encapsulates multiple operational objectives and constraints that are critical for modeling the ship scheduling challenges across the canal. Notably, the parameters in this model are highly entangled and exhibit significant sensitivity—small adjustments can lead to pronounced variations in the resulting schedule. This high sensitivity is especially critical during drought seasons when water scarcity demands tighter control over resource usage, and when the benefit parameters—representing the inherent prioritization of ships—must be balanced precisely to reflect the operational imperatives.

Due to the highly sensitive effect parameters, and the need to enhance the adaptability and performance of our scheduling system, we propose an integrated parameter tuning framework based on reinforcement learning (RL). This AI approach is essential due to the highly entangled nature of the parameters and their extreme sensitivity to the results. Additionally, the framework is designed to accommodate variability in operational conditions—such as increased water scarcity during drought seasons and dynamic shifts in ship prioritization as embedded in the benefit parameters. The RL agent dynamically adjusts the QUBO penalty and reward coefficients to optimize the balance between water cost minimization and timetable feasibility.

5.2. AI Implementation

The proposed algorithmic steps are as follows:

1. **Initialization:** Start with an initial set of QUBO parameters.
2. **Timetable Generation:** Use an optimization method (e.g., quantum annealing, simulated annealing, or genetic algorithms) to generate a candidate timetable from the current QUBO formulation.
3. **Evaluation:** Assess the candidate timetable by computing its total water usage, the incurred costs, and any infeasibility issues (such as over-assignment or capacity violations).
4. **Reward Formation:** Construct a composite reward signal that integrates water cost savings with a feasibility score, and which reflects the balance between the penalty contributions and the benefit parameters.
5. **Parameter Update:** Employ a reinforcement learning algorithm (such as Q-learning or policy gradient methods) to update the QUBO parameters based on the cumulative reward feedback.
6. **Iteration:** Repeat the timetable generation, evaluation, and parameter update steps until convergence is achieved, yielding an optimized set of coefficients tuned to current operational conditions.

The dataset underpinning our study is extracted from historical transit records and operational data provided by the Panama Canal Authority. It includes detailed information on ship characteristics (lengths, benefits, and types), lock specifications (capacities and baseline water costs), and time slot availability. The inherent variability of seasonal factors such as drought-induced water restrictions and dynamically shifting priorities in ship assignments—further justifies our choice of a reinforcement learning approach. By continually recalibrating system parameters in response to real-time performance metrics, the RL-based framework ensures a judicious balance between minimizing water consumption and enforcing operational feasibility.

6. Quantum Computing Integration & Feasibility

6.1. Quantum Adiabatic Algorithm (QAA)

In this proposal, we deploy a Quantum Adiabatic Algorithm (QAA) to solve the Quadratic Unconstrained Binary Optimization (QUBO) problem. Although both QAOA and QAA have been considered, the design here emphasizes an adiabatic evolution approach, which is naturally suited for fully analog neutral atom platforms.

If implemented exactly, iterative quantum methods can evolve the system toward low-energy states representing water-efficient configurations. However, achieving an exact Hamiltonian on neutral atom platforms is challenging because it requires positioning atoms to satisfy a large, quadratically growing set of constraints,[18] leading to only approximate implementations. Additionally, the low repetition rates (around 1–5 Hz) of current devices mean that obtaining statistically precise energy measurements is time-consuming and makes QAOA practically infeasible for large implementations.

The QAA works by initializing the system in the ground state of an easily prepared Hamiltonian H_0 and then evolving the Hamiltonian slowly to a final problem Hamiltonian H_P that encodes the QUBO. The adiabatic theorem guarantees that if the evolution is slow compared to the inverse of the square of the minimum energy gap, the system remains in (or near) the ground state, ideally ending in the solution state of the QUBO.

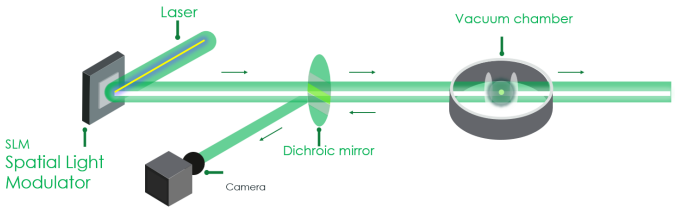

Figure 6. Neutral Atom Quantum Computer *(Image by Pasqal)*

6.2. Main Algorithmic Steps

We start with an initial ground state Hamiltonian, $H(0)$ which corresponds to the initial state $|0..0\rangle$. The problem QUBO is encoded as an Ising Hamiltonian. The diagonal elements of the QUBO provide local bias terms (nonuniform local fields) and off-diagonals define the interactions:

$$H_P = \sum_i h_i \, \sigma_i^z + \sum_{i<j} J_{ij} \, \sigma_i^z \sigma_j^z,$$

where h_i corresponds to the diagonal entries (e.g., 95, −55, etc.) and J_{ij} is derived from the off-diagonal elements.The system Hamiltonian is interpolated as:

$$H(s) = (1-s)\,H_0 + s\,H_P, \quad s = \frac{t}{T}, \quad t \in [0, T].$$

A custom pulse sequence is designed to implement this interpolation. Due to the nonuniform local fields, each qubit (atom) must have an individually tailored detuning ramp, i.e., for each atom i, the detuning $\delta_i(t)$ is set such that $\delta_i(T) \approx h_i$.

A series of time-dependent pulses is defined to drive the system from H_0 to H_P. The sequence is discretized into small time steps, each represented by a pulse with specific amplitude, phase, and duration. To simulate the individual detuning ramps and interactions, a virtual device is implemented. This allows individual control of the detuning parameters for each atom, and customized engineering of the interaction pulses to reflect the QUBO's coupling terms. Finally The designed sequence is simulated using the Qutip emulato. After the adiabatic evolution, the final quantum state is measured. The measurement outcome (bitstring) corresponds to the solution of the QUBO problem (e.g., optimal crossing schedule for the Panama Canal).

6.3. Parameters and Their Calculation

The total annealing time T must be chosen based on the minimum energy gap encountered during evolution. Longer T improves adiabaticity but may be limited by decoherence. Local Detuning $\delta_i(t)$ for each atom i is designed such that:

$$\delta_i(t) = f_i(t), \quad \text{with } \delta_i(T) \approx h_i,$$

where $f_i(t)$ is a function that is calibrated against the QUBO diagonal bias.

Each pulse in the sequence has parameters (amplitude, duration, phase) which are determined by T=the required change in the Hamiltonian at each time step, and the calibration against the physical constraints of the neutral atom system. The interaction strengths J_{ij} are directly taken from the QUBO off-diagonal entries, rescaled to match the hardware's native interaction range.

6.4. Rationale Behind Pulse and Channel Selection

The global channel applies the Rydberg driving field across all atoms simultaneously.For the global rydberg channel driving pulse we use a Blackman Waveform because of its smooth temporal profile and controlled decay. The Blackman waveform is selected for its smooth edges and low spectral leakage. This is essential for reducing unnecessary transitions and ensuring that the Rydberg excitation is applied uniformly across all atoms. The waveform gradually reduces the driving amplitude from 15 to 0 rad/μ s over the pulse duration. This controlled decay helps in maintaining the adiabaticity of the evolution by slowly "turning off" the global drive.

The local channel has its the amplitude is fixed at zero. This design choice ensures that the channel only modifies the detuning (i.e., the energy offset) without inadvertently driving any transitions, keeping the control solely on the local field configuration. For the local detuning pulses, we use a Ramp Waveform because of its gradual detuning change and tailored control.The Ramp Waveform is used for per-atom detuning control. It provides a linear (or customizable) ramp that smoothly transitions the detuning from an initial value (typically 0) to the target value extracted from the QUBO's diagonal entries. This ensures that each atom receives a detuning profile matching its required bias, which is critical for accurately mapping the QUBO problem onto the physical system.

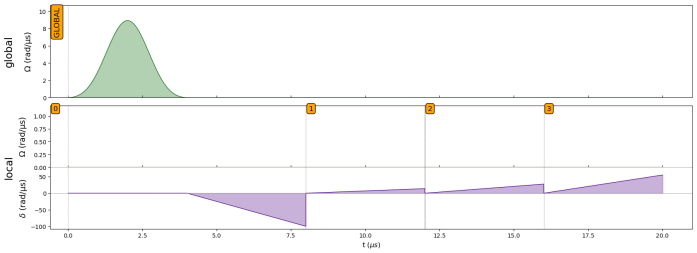

Figure 7. Pulse configuration for 2 ship 2 time slot QUBO

6.5. Feasibility and Scalability

The problem scales on qubits based on the number of ships and number of time slots i.e. the number of qubits required is equal to N*T. Analysis shows that with current-generation devices (up to 100 qubits) the approach is feasible only on small QUBO's of instance sizes of 10 ships and 10 time slots. Next-generation devices with partial addressability (targeting up to 400 qubits) will allow for larger problems. For the Panama Canal simulation, where an average of 32 crossings may require around 1024 qubits, the problem can be partitioned into sections of the day to reduce the number of qubits required at one time.The addressability of each qubit in current and next generation quantum computers does possess an issue as our current implementation requires each qubit to have custom detuning.

6.5.1. Removing Detuning Requirement

However, we can get around this by assuming the diagonal is uniform. While some information is lost within this implementation, in can be reconciled. The non-uniform parts of the diagonal correspond to solutions with ships that are too long for the Panamax locks, therefore we can just evaluate the solutions and neglect not feasible one and remove the requirement for detuning. This is the method that will be implemented for the results.

Moreover, because the diagonal terms remain identical, a global Rydberg beam with a well-chosen detuning Δ can be used. Otherwise, local addressing capabilities would be required. Now the Hamiltonian is given by

$$H_Q = \sum_{i=1}^{N} \frac{\hbar\Omega}{2} \, \sigma_i^x - \sum_{i=1}^{N} \frac{\hbar\delta}{2} \, \sigma_i^z + \sum_{j<i} \frac{C_6}{\left|\mathbf{r}_i - \mathbf{r}_j\right|^6} \, n_i n_j \,, \qquad (12)$$

where Ω represents the Rabi frequency, δ the detuning, and C_6 characterizes the van der Waals interaction between atoms located at positions $\mathbf{r}_i$ and $\mathbf{r}_j$. The term $n_i n_j$ accounts for the excitation occupations at the respective sites.

We continuously adjust the control parameter Ω, δ over time, starting at $\Omega_i = 0, \delta < 0$ and progressing to $\Omega_f = 0, \delta > 0$. Here, the ground state of $H(\Omega_i)$ coincides with our initial state, while the ground state of $H(\Omega_f)$ corresponds to the target ground state of H. To ensure that the system does not get excited into undesirably high-energy states, we keep the parameter $\Omega \in [0, \Omega_{max}]$ fixed, and we select Ω_{max} as the median of the available Q-values. This choice guarantees that the adiabatic pathway remains

7. Method

7.1. Simulated Implementation

7.1.1. Process

The mathematical formulation is tested using a simulated annealing sampler from the dimod library, it is a classical CPU heuristic solver that mimic the physical process of annealing, starting from a random configuration and "cools" the system to slowly to settle into a low-energy (i.e. near optimal) configuration. Running this sampler returns solution samples with the associated "energy" (objective value as computed by the QUBO).

Values for ships lengths are generated based on the distribution of ships that currently pass through the Panama Canal. A discrete number of time slots are made where there are dual Panamax time slots and NeoPanamax time slots. Ships are assigned to these time slots based on our QUBO which is minimised. The QUBO is minimised a number of times and the best solution is selected via the post-processing. A better formulated QUBO reduces the number of iterations needed to find the optimal solution, but as before the infinite possible reward and penalty require a number of QUBO iterations to be run in order for the most optimal solution to be found.

Post-processing is then performed on each candidate solution by enforcing constraints. The water cost was calculated by for each of the remaining solutions, cross filling is applied by reducing the water usage of the next Panamax lock by some factor in our case 0.3. The optimised water cost is the sum of all used water slots, this results in a lower overall water usage if tandem lockages were applied. In order to calculate a baseline water cost to compare against, we sum over the over all-time slots are the lock water usage, this represents the current usage at the Panama Canal where one boat passes through at a time and no water saving measures are applied. In order to select the best timetable, the total "energy" is computed as the negative of the total benefit plus the total penalty and the weighted total water cost. Computing the total "energy" allows use to manage prioritisation of ships to manage which ships are allowed through in the event more ships are awaiting crossing than can possibly cross at such a time.

7.1.2. Simulation Details

Parameters are chosen in by systematically by managing the number of unfeasible solutions. As mention before "hard" constraint can be implemented where the value of the penalty far outweighs any possible reward for a solution, however we found better results using "soft" constraints where infeasible solutions are simply discouraged rather than forbidden and then post-processing any infeasible solutions after Simulated Annealing is completed on a QUBO.

To assess the computational complexity of evaluating our QUBO, we generate all possible bitstrings of length n (where n is the dimension of the QUBO). For each bitstring, we compute the cost using the quadratic form

$$C(\mathbf{z}) = \mathbf{z}^\top Q \mathbf{z},$$

which requires $O(n^2)$ operations. Since there are 2^n possible bitstrings, the overall time complexity is modeled as:

$$T(n) = C \cdot \left(2^n \cdot n^2\right),$$

where C is a constant scaling factor. To calibrate the constant C, we use the normalized simulated time for the smallest instance size n_0.

Parameter	Value
λ_{ship}	20
$\lambda_{\text{conflict}}$	10
λ_{water}	0.1
λ_{length}	20
λ_{tandem}	10.0
$\lambda_{\text{crossfill}}$	10.0
Crossfill factor	0.3
NUM_READS	50

Table 1. Parameter Values for Optimization Model

That is,

$$C = \frac{\text{normalized\_simulated\_time[0]}}{2^{n_0} \cdot n_0^2}.$$

Using this scaling factor, the theoretical computation time per run for any instance size n is then given by:

$$T(n) = C \cdot \left(2^n \cdot n^2\right).$$

7.2. Quantum Adiabatic Algorithm (QAA) Implementation

The Quantum Adiabatic Algorithm (QAA) is employed to solve the QUBO problem by adiabatically evolving a system of neutral Rydberg atoms from an easily prepared initial ground state to the ground state of the problem Hamiltonian H_Q. To encode the QUBO matrix Q onto the neutral-atom hardware, the off-diagonal elements of Q are mapped to the Rydberg interaction between atoms. The interaction between atoms i and j is given by:

$$U_{ij} = \frac{C_6}{r_{ij}^6},$$

where r_{ij} is the distance between the atoms and C_6 is the van der Waals coefficient. A minimization routine is used to adjust the atomic coordinates in the register so that the resulting interaction matrix approximates the off-diagonal terms of Q as closely as possible.

7.2.1. Design of the Adiabatic Pulse

To simplify the Pulse calculation, we transform the diagonals such that they are all the same value. This modifies our solution space o include infeasible solutions, but we simply iterate over the solution bitstrings until we find the first feasible solution. This solution still represents the minimal water usage because the QUBO terms we change just include penalties from the length constraint. After embedding the QUBO, the system is evolved using an adiabatic pulse that continuously varies the control parameters. The time-dependent Hamiltonian is governed by two key parameters: the Rabi frequency $\Omega(t)$ and the detuning $\delta(t)$. The evolution is designed with the following boundary conditions:

$$\Omega(0) = 0, \quad \Omega(T) = 0, \quad \delta(0) < 0, \quad \delta(T) > 0,$$

where T is the total evolution time. In our implementation, Ω is chosen as the median of the positive off-diagonal elements of Q, ensuring an efficient adiabatic path. The interpolated pulse is constructed using Pulser's `InterpolatedWaveform` as shown below:

```
adiabatic_pulse = Pulse(
InterpolatedWaveform(T, [1e-9, Omega, 1e-9]),
InterpolatedWaveform(T, [delta_0, 0, delta_f]),
0
)
```

Code 1. QAOA conversion code

Here, δ_0 is a negative detuning and δ_f is the corresponding positive detuning.

7.2.2. Simulation of the QAA Evolution

The adiabatic pulse is added to a sequence that is executed on a simulated neutral-atom device using Pulser's Qutip emulator. The sequence is created by first declaring a global Rydberg channel and then adding the designed pulse:

```
seq = Sequence(reg, DigitalAnalogDevice)
seq.declare_channel("ising", "rydberg_global")
seq.add(adiabatic_pulse, "ising")
simul = QutipEmulator.from_sequence(seq)
results = simul.run()
count_dict = results.sample_final_state()
```

Code 2. QAOA conversion code

The final state is sampled to obtain a distribution of bitstrings, from which the optimal solutions to the QUBO problem are inferred.

7.2.3. Analysis of Time Evolution Effects

To assess the impact of the total evolution time T on the performance of the QAA, we perform a series of simulations varying T over a range of values. For each simulation, the cost of a sampled bitstring $\mathbf{z}$ is computed as:

$$C(\mathbf{z}) = \mathbf{z}^\top Q \mathbf{z}.$$

The average cost is calculated by averaging over the samples, and the resulting cost is plotted against the evolution time T to determine the optimal duration for adiabatic evolution.

Overall, this QAA method demonstrates rapid convergence to the ground state, achieving high-quality solutions for the QUBO instance in only a few microseconds of evolution time.

8. Results

8.1. Simulated Results

Instance size refers to the number of ships, in these simulations this is equivalent to the number of time slots, the instance size squared refers to the qubit size needed. As the instance size increases, the water cost tends to grow due to the increased complexity of the scheduling problem. However, when compared to the baseline water usage (where only single-ship lockages are allowed), the implementation of water-saving measures—such as cross-filling and tandem lockages—yields a significant reduction in overall water consumption.

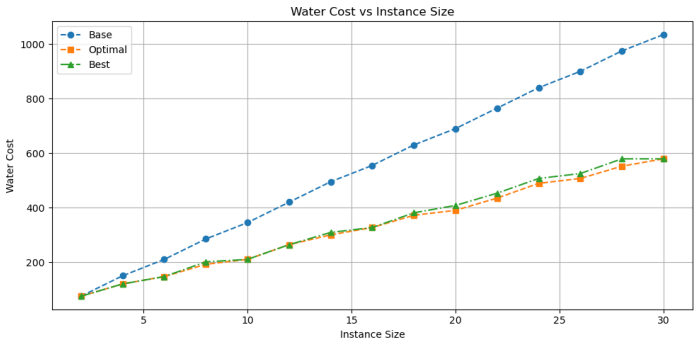

Figure 8. Water Cost vs Instance Size

We also see that our Simulated Results almost directly map to the optimal results that are calculated classicly this implies are parameters have been adopted successfully. The benefit out our QUBO implementation is that we find the solution more quickly than the classical way.

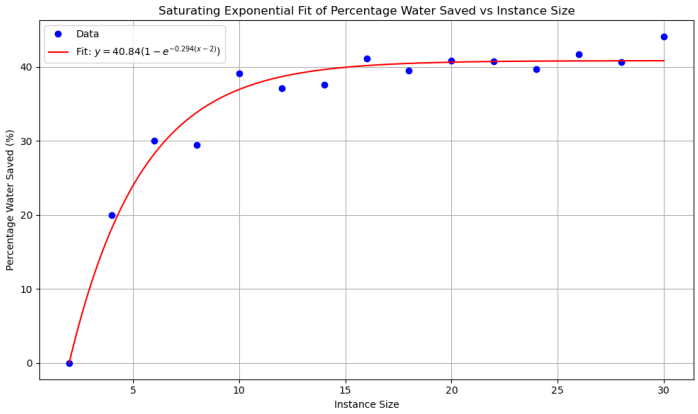

Figure 9. Percentage of Water vs Instance Size

Beyond an instance size of 10 we have an average of 40% of water saved in comparison to baseline usage, which is the method of operation in place currently at the Panama Canal.There is a more gradual reduction in water cost. There are slight fluctuations along the way—a characteristic feature of its probabilistic method for escaping local minima. Despite these oscillations, SA consistently trends downward. These minor oscillations reflect SA's exploration mechanism, which, while less aggressive in early convergence, ensures robustness by avoiding premature lock-in to suboptimal solutions.

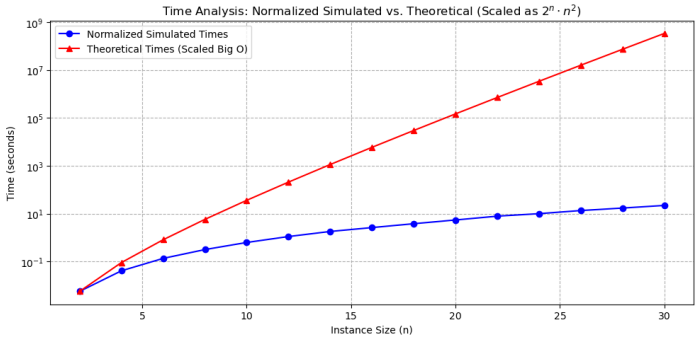

Figure 10. Time for Simulated Annealing vs. Classical

In addition, we compare the computational time required for simulated annealing against classical optimization methods. The results indicate that the annealing-based approach not only converges to near-optimal solutions but also scales competitively as the problem size increases.

8.2. QAA Results

8.3. Embedding

Quantum Adiabatic Algorithm (QAA) experiments were conducted to further validate our QUBO formulation. Figure **??** displays the embedding configuration obtained for a 4-qubit instance, demonstrating the feasibility of mapping the problem onto a neutral atom quantum processor. In Figure **??**, the embedding time is plotted against the number of qubits, suggesting that while the current implementation is efficient for small-scale instances, scaling to larger problems will require next-generation hardware.

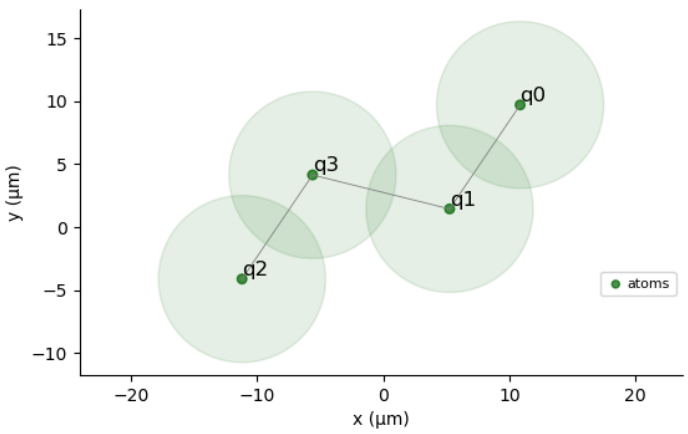

Figure 11. Embedding used for 4 qubit Results

The embedding graph illustrates the spatial arrangement of qubits for a 4-qubit instance, demonstrating how the QUBO problem is physically mapped onto a quantum processor. This mapping ensures that atomic interactions—governed by Rydberg blockade effects—mirror the couplings in the QUBO matrix.

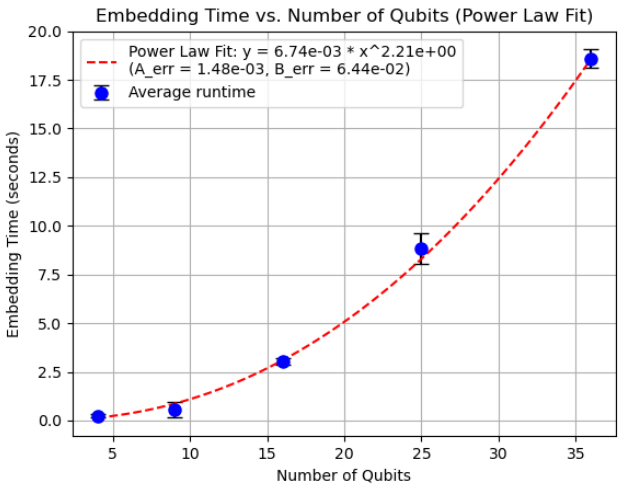

Figure 12. Embedding Time vs Number of Qubits

However, Figure 12 reveals a critical scalability limitation: embedding time grows polynomially with qubit count, following a power-law relationship. This increase arises because arranging atoms to satisfy the QUBO's interaction constraints becomes exponentially more complex as the number of qubits (and thus interactions) expands. For instance, doubling qubits quadruples potential couplings, demanding meticulous optimization of atomic positions to avoid physical conflicts (e.g., overlapping blockade radii). While current hardware handles small instances (e.g., 4–10 qubits) efficiently, scaling to practical problem sizes (e.g., 32 crossings requiring 1024 qubits) would require embedding times that grow prohibitively long under existing methods. This underscores the need for next-generation neutral atom platforms with higher qubit densities, improved local control, and advanced embedding algorithms to mitigate these bottlenecks.

8.4. QAA

We illustrate the pulse sequence used in the 4-qubit QAA experiment. The smooth Blackman waveform (for the global channel) combined with a Ramp waveform for local detuning provides a controlled evolution of the system. This is confirmed by the time evolution of the Hamiltonian cost, where a gradual decrease in cost indicates convergence towards a low-energy, near-optimal solution.

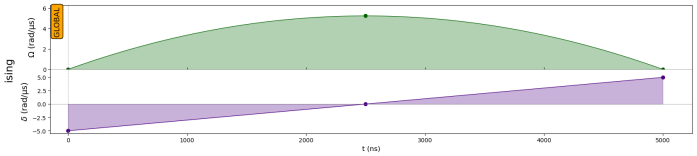

Figure 13. Pulse Used for 4 qubit

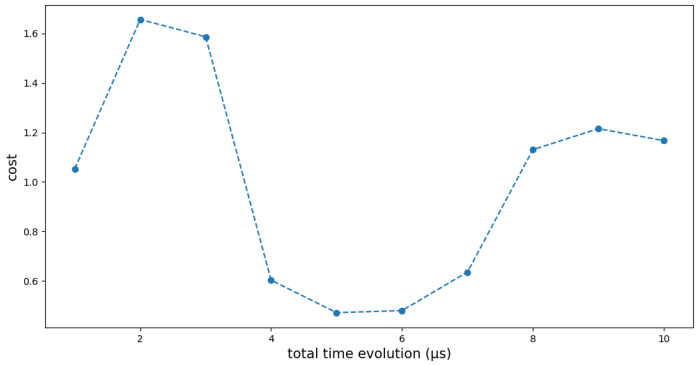

Figure 14. Time Evolution of Hamiltonian Cost

The graph for QAA shows an early steep decline in the water cost value. This rapid drop suggests that QAA quickly identified promising regions in the solution space and converged smoothly to the optimal zone. Once near the minimum, the curve levels off, implying limited further improvements and stable convergence. This smooth plateau indicates that QAA is highly effective at honing in on high-quality solutions early in the search process.

The experimental results demonstrate that both Quantum Adiabatic Algorithm (QAA) and Simulated Annealing (SA) successfully generated timetables with minimal water costs, validating the efficacy of the QUBO formulation in optimizing Panama Canal scheduling under water scarcity.

The QAA experiments, though limited by current hardware, align with SA outcomes. Figure 11 depicts a 4-qubit embedding, confirming feasibility on neutral atom processors. However, Figure 12 highlights a power-law increase in embedding time with qubit count, suggesting scalability challenges for large instances. The pulse sequence (Figure 13) and Hamiltonian cost evolution (Figure 14) show controlled adiabatic evolution, with energy decreasing as the system approaches low-energy states. Notably, QAA produced identical timetables to SA after post-processing infeasible solutions, verifying algorithmic correctness.QAA matches SA results but requires hardware advancements (e.g., qubit count, detuning precision) to outperform classical methods at scale.

8.5. Roadmap and Future Work

Our current work has established a robust framework for applying QUBO-based optimization to the Panama Canal scheduling problem under water scarcity conditions. To bridge the gap between our theoretical model and real-world implementation, it is essential to enhance the QUBO formulation so that it more accurately reflects the intricacies of canal operations. In the near term, our efforts will focus on integrating additional operational constraints, such as dynamic weather patterns, maintenance schedules, variable lock operation times, and finer water consumption metrics. By refining the penalty and reward terms—potentially through adaptive tuning via reinforcement learning—we aim to capture the complex trade-offs present in the actual operational environment.

8.6. Errors in QAA on Nuetral atoms

There are several error sources intrinsic to the hardware and methodology. First, deviations from the adiabatic condition—due to finite evolution times or non-ideal control of Hamiltonian parameters—risk exciting the system into higher-energy states, corrupting the solution. Second, decoherence from environmental noise (e.g., thermal fluctuations or stray electric fields) limits coherence times, particularly for long-duration annealing processes. Third, imperfect Rydberg interactions, such as crosstalk or miscalibrated blockade radii, distort the intended coupling strengths in the QUBO matrix, leading to incorrect problem encoding. Additionally, positioning inaccuracies during atom trapping may misalign interaction distances, further degrading the fidelity of mapped constraints. Finally, measurement errors during readout, caused by finite detection efficiency or state leakage, introduce uncertainty in solution verification.

Significant improvements are required in PASQAL's neutral atom technology to fully exploit its potential for quantum annealing and optimization tasks. In particular, enhancements in detuning capabilities are essential; current systems must advance to allow for highly precise, per-atom detuning control with minimal cross-talk, enabling each qubit's energy level to be individually tuned with greater accuracy. Moreover, reducing the minimum distance between atoms is critical for scaling up the quantum register, as a denser atomic array would facilitate larger problem embeddings while maintaining the integrity of the Rydberg blockade. Additional technological advances, such as improved laser stabilization, enhanced optical resolution, and more robust error mitigation techniques, are also necessary to ensure reliable and homogeneous control over the qubits. Together, these improvements will help bridge the gap between existing neutral atom implementations and the demands of large-scale, high-fidelity quantum computations.

Despite these promising directions, several challenges remain. The exponential complexity of evaluating all possible bitstrings, quantified as $O(2^n \cdot n^2)$, represents a significant computational bottleneck that may necessitate further research into approximation and problem decomposition techniques. Additionally, the successful integration of high-quality, real-time operational data is critical yet challenging. Current hardware limitations, particularly in terms of qubit count and noise, also pose barriers to scaling. However, as these challenges are gradually addressed through both technological advancements and algorithmic innovations, our roadmap provides a clear trajectory from enhanced model development and pilot testing to full-scale deployment in the FTQC era, ultimately promising substantial improvements in operational efficiency and water conservation.

9. Conclusion

In this work, we developed a comprehensive framework to address the Panama Canal scheduling problem under water scarcity conditions by formulating it as a Quadratic Unconstrained Binary Optimization (QUBO) problem. Our approach successfully integrates water-saving measures such as cross-filling and tandem lockages, which were shown to reduce overall water consumption when compared to baseline operations.

Classical simulated annealing experiments demonstrated that our QUBO formulation converges to near-optimal solutions with competitive computational times as the instance size increases. Moreover, our proof-of-concept quantum annealing (QAA) experiments—implemented on a neutral atom quantum processor—highlight the potential of quantum techniques to tackle such complex, real-world optimization challenges. The embedding results and pulse sequence configurations further support the feasibility of mapping the scheduling problem onto a quantum platform.

The incorporation of reinforcement learning for dynamic QUBO parameter tuning provides promise in balancing conflicting objectives, thereby enhancing both the feasibility and efficiency of the generated timetables. Although current quantum hardware restricts us to small-scale instances, our results indicate that scaling to larger problems is plausible with next-generation devices and advanced embedding techniques.

Overall, this study provides a viable and scalable strategy for improving water efficiency in critical infrastructure management. Future work will focus on refining the parameter tuning process, mitigating hardware limitations, and extending the framework to handle larger, more complex instances. The insights gained from this work lay a solid foundation for further exploration into quantum-enhanced optimization methods in operational research.

10. Appendix

10.1. GitHub

Github. Repository for Code and Sample Timetables generated.

10.2. References

[1] Embassy of Panama. *Panama Canal*. Accessed: 2025-01-28. 2023. URL: https://www.embassyofpanama.org/panama-canal.

[2] Panama Canal Authority. *How the Panama Canal Is Addressing the Issue of Water Head-On*. Accessed: 2025-01-28. 2023. URL: https://pancanal.com/en/how-the-panama-canal-is-addressing-the-issue-of-water-head-on.

[3] Scope of Work. *Panamaximization*. Accessed: 2025-01-28. 2025. URL: https://www.scopeofwork.net/panamaximization/.

[4] United Nations. *Sustainable Development Goals*. Accessed: 2025-03-08. 2025. URL: https://sdgs.un.org/goals.

[5] Leonid Ponomarenko, Aleksandr I. Tikhomirov, et al. "Probing the limits of numerical evaluation of statistical properties of chaotic systems". In: *arXiv preprint arXiv:1601.03030* (2016). URL: https://arxiv.org/abs/1601.03030.

[6] Michael Streif and Martin Leib. "Comparison of QAOA with Quantum and Simulated Annealing". In: *arXiv preprint arXiv:1901.01903* (2019). URL: https://arxiv.org/abs/1901.01903.

[7] Sergey Knysh Hidetoshi Nishimori Junichi Tsuda. "Comparative Study of the Performance of Quantum Annealing and Simulated Annealing". In: *arXiv preprint arXiv:1409.6386* (2014). URL: https://arxiv.org/pdf/1409.6386.

[8] Jehn-Ruey Jiang and Chun-Wei Chu. "Solving NP-hard Problems with Quantum Annealing". In: *Journal of XYZ* 10.3 (2022). Accessed: 2025-02-15, pp. 123–456. URL: https://staff.csie.ncu.edu.tw/jrjiang/publication/T220032_Final.pdf.

[9] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. "A quantum approximate optimization algorithm". In: *arXiv preprint arXiv:1411.4028* (2014).

[10] Elizabeth Crosson and Aram W. Harrow. "Simulated quantum annealing can be exponentially faster than classical simulated annealing". In: *2016 IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2016, pp. 714–723.

[11] Fred Glover and Gary Kochenberger. "A Tutorial on Formulating and Using QUBO Models". In: *arXiv* (2019). Submitted on 13 Nov 2018, last revised 4 Nov 2019. DOI: 10.48550/arXiv.1811.11538.

[12] Nikola Šibalić and Charles S Adams. *Rydberg physics*. IOP Publishing, 2018.

[13] Mikhail D Lukin et al. “Dipole blockade and quantum information processing in mesoscopic atomic ensembles”. In: *Physical Review Letters* 87.3 (2001), p. 037901.

[14] Kai-Niklas Schymik et al. “In situ equalization of single-atom loading in large-scale optical tweezer arrays”. In: *Physical Review A* 106.2 (Aug. 2022).

[15] Edoardo Alessandroni et al. “Alleviating the quantum Big-*M* problem”. In: *arXiv preprint arXiv:2307.10379* (2023). eprint: 2307.10379. URL: https://arxiv.org/abs/2307.10379.

[16] Lennart Bittel and Martin Kliesch. “Training Variational Quantum Algorithms Is NP-Hard”. In: *Physical Review Letters* 127 (12 Sept. 2021), p. 120502. DOI: 10.1103/PhysRevLett.127.120502.

[17] Wei Zhang, Ming Li, and Jun Wang. “A New Approach to Solving Linear Systems”. In: *Applied Mathematics and Computation* 394 (2022), p. 125678.

[18] M. Cerezo et al. “Variational quantum algorithms”. In: *Nature Reviews Physics* 3.9 (2021), pp. 625–644.