Implementación de QAOA en el Problema de Reasignación de Puestos de Trabajo: Un Análisis Empírico

Adriano Lusso[1], Christian Nelson Gimenez[1], and Alejandro Mata Ali[2]

[1] Universidad Nacional del Comahue, Neuquén, Argentina
christian.gimenez@fi.uncoma.edu.ar, adriano.lusso@est.fi.uncoma.edu.ar
[2] Instituto Tecnológico de Castilla y León, Burgos, España
alejandro.mata@itcl.es

Abstract. En la última década, se ha logrado un progreso significativo en el desarrollo de computadoras NISQ (*Noisy Intermediate-Scale Quantum*), aunque se requieren mejoras en el hardware para que los algoritmos cuánticos a gran escala se ejecuten sin errores. Mientras tanto, los investigadores continúan enfocándose en el desarrollo de algoritmos efectivos para el hardware actual, con énfasis en aplicaciones a corto plazo como la optimización combinatoria. Este estudio presenta un análisis comparativo del Algoritmo Cuántico de Optimización Aproximada (QAOA) aplicado al problema de Reasignación de Puestos de Trabajo (JRP), que consiste en asignar n trabajadores a m trabajos vacantes para maximizar la realización de tareas de alta prioridad y la satisfacción de los trabajadores con sus asignaciones. El análisis, realizado mediante la simulación clásica en 105 instancias de JRP, indica un resultado prometedor, con razones de aproximación notablemente altas que oscilan principalmente entre $0,86$ y $0,97$. Esto contribuye a tener un incremento medio del 12% en la productividad organizacional, al mejorar la asignación de tareas de alta prioridad y la satisfacción de los trabajadores con sus asignaciones.

Keywords: Reasignación de Puestos de Trabajo, QAOA, Aplicación al Mundo Real, Benchmarking, NISQ

Benchmarking QAOA on the Job Reassignment Problem: An Empirical Analysis

Adriano Lusso[1], Christian Nelson Gimenez[1], and Alejandro Mata Ali[2]

[1] Universidad Nacional del Comahue, Neuquén, Argentina
christian.gimenez@fi.uncoma.edu.ar, adriano.lusso@est.fi.uncoma.edu.ar
[2] Instituto Tecnológico de Castilla y León, Burgos, España
alejandro.mata@itcl.es

Abstract. In the past decade, there has been significant progress in the development of NISQ (Noisy Intermediate-Scale Quantum) computers, though further hardware improvements are necessary for large-scale quantum algorithms to execute without errors. In the meantime, researchers continue to focus on developing effective algorithms for current hardware, with an emphasis on near-term applications like combinatorial optimisation. This study presents a benchmarking analysis of the Quantum Approximate Optimisation Algorithm (QAOA) applied to the Job Reassignment Problem (JRP), which involves assigning n workers to m vacant jobs to maximize high-priority task completion and worker satisfaction. The benchmarking, performed with classical simulation on 105 JRP instances, shows promising results with approximation ratios ranging from 0.86 to 0.97. This leads to an average improvement of 12% in the organisational productivity thanks to a better assignment of high-priority tasks and worker satisfaction.

Keywords: Job Reassignment Problem, QAOA, Real-world Application, Benchmarking, NISQ

1 Introduction

The advent of Noisy Intermediate-Scale Quantum (NISQ) computers has sparked tremendous interest in the field of quantum computing (De Luca, 2022). Although these devices are still in the early stages of development, they hold the potential to address a wide range of applications with improved space and time complexity. However, the limitations of current quantum hardware —such as noise, decoherence, and error rates— pose significant challenges for running large-scale, fault-tolerant quantum algorithms. As a result, a major focus in quantum computing research is the design of algorithms that can operate effectively on smaller quantum processors, often with only a few hundred qubits, while preparing the ground for future developments in Fault-Tolerant Quantum Computing (FTQC).

One of the most promising areas of exploration is combinatorial optimisation, where quantum computing techniques can offer substantial advantages over classical methods (Gemeinhardt et al., 2023). The Quantum Approximate Optimisation Algorithm (QAOA) (Farhi et al., 2014), along with other quantum approaches such as Quantum Annealing (Rajak et al., 2022) and the Quantum Variational Eigensolver (QVE) (Tilly et al., 2022), has emerged as a feasible tool for solving hard optimisation problems. These algorithms leverage the principles of quantum mechanics to explore a large solution space efficiently, offering the potential to find optimal or near-optimal solutions in scenarios where classical algorithms face exponential time complexity.

Among the numerous combinatorial problems that have been targeted by quantum algorithms (Abbas et al., 2024), the Job Reassignment Problem (JRP) stands out as an important real-world application (Delgado et al., 2023). The JRP involves assigning workers to vacant job positions in such a way that the overall completion of high-priority tasks is maximised, while also ensuring that the workers are satisfied with their assignments. While these workers are already assigned to specific tasks, an optimal configuration could also involve some of them not being reassigned.

In this study, we focus on simulating QAOA applied to the JRP in a noiseless environment. The JRP can be viewed as a special case of the classic Assignment Problem (Pentico, 2007), which already has efficient classical solvers such as the Hungarian method and integer programming (Kuhn, 1955; Papadimitriou & Steiglitz, 1998). Its well-defined structure and known classical behaviour make it a compelling candidate for benchmarking quantum heuristics, providing a clear baseline for evaluating performance under NISQ-era constraints.

Our motivation for exploring QAOA in this context is twofold. First, applying QAOA to archetypal combinatorial tasks such as assignment problems provides valuable insights into its performance and scaling under the typical constraints of NISQ devices, including limited qubit count, shallow circuit depth and a restricted number of optimization iterations. Second, while the classical formulation of the JRP is tractable, real-world variants often incorporate soft constraints (Hmer & Mouhoub, 2010; Meseguer et al., 2006), features that QAOA can naturally encode through its variational structure (Hadfield et al., 2017), whereas classical exact solvers may struggle to integrate these without incurring significant computational overhead (Cohen et al., 2004).
Our main contributions are the following:

- We evaluate **QAOA on 105 JRP instances** under noiseless simulation, exploring its effectiveness across varying **problem sizes and QAOA depths (p)** on solution quality.
- We assess **practical utility** using a gain improvement metric, showing benefits over non-reassignment scenarios.
- We analyse **energy differences**, highlighting how instance structure and hyperparameters influence QAOA performance.
- We discuss the influence of **TQA-inspired initialization**, showing how it may guide more effective parameter selection in deeper QAOA circuits.

This paper is structured as follows. Section 2 presents the Job Reassignment Problem and its formulation. Section 3 describes the benchmarking process, including the experiment design and setup. Section 4 reports and discusses the results. Finally, Section 5 concludes the paper and outlines directions for future work. A repository with the implementation for this paper is available at https://github.com/AdrianoLusso/QuantumComputing_for_JobReassignmentProblem.

2 Job Reassignment Problem

The Job Reassignment Problem (JRP) involves an organisation with J workers, each of whom is assigned a job (Delgado et al., 2023). Due to unforeseen circumstances, I new high-priority vacant positions are created. The objective is to identify the workers who are best suited for the I vacant positions and perform the reassignment. To accomplish this, a total of $I \cdot J$ scores, denoted S_{ij}, are considered. These scores represent the degree to which worker j is suited to vacant job i. The optimal configuration is the one that maximises the sum of the scores S_{ij}.

2.1 Scores S_{ij} definition

Each S_{ij} in the gain function consists of the Priority Gain $\Delta^{\mathcal{P}}_{ij}$ and the Affinity Gain $\Delta^{\mathcal{A}}_{ij}$.

The Priority Gain is defined as $\Delta^{\mathcal{P}}_{ij} \equiv \mathcal{P}^{V}_{i} - \mathcal{P}^{C}_{j}$, where the priority of the vacant job is $\mathcal{P}^{V}_{i} \in (0, 1]$ and the priority of the job currently assigned to the worker is $\mathcal{P}^{C}_{j} \in (0, 1]$. In the problem domain, the priority value of any job is defined by its importance for the productivity and efficiency of the organisation.

The Affinity Gain is defined as $\Delta^{\mathcal{A}}_{ij} \equiv \mathcal{A}^{V}_{ij} - \mathcal{A}^{C}_{jj}$, where the personal affinity of worker j with vacant job i is $\mathcal{A}^{V}_{ij} \in (0, 1]$, and the personal affinity of the worker with the job currently assigned is $\mathcal{A}^{C}_{jj} \in (0, 1]$. In the problem domain, the affinity value of a worker with a job characterises the motivation and job satisfaction generated by the worker being assigned to that position. Various factors, such as distance from home, job difficulty, and salary, can affect this value.

Finally, Eq. (1) defines the total score S_{ij} for worker j and vacant job i. The **gain coefficients** $c^{\mathcal{P}}$ and $c^{\mathcal{A}}$ are defined as positive constants that quantify the relative weight of each term.

$$S_{ij} = c^{\mathcal{P}} \Delta^{\mathcal{P}}_{ij} + c^{\mathcal{A}} \Delta^{\mathcal{A}}_{ij}. \tag{1}$$

2.2 QUBO Formulation

According to Delgado et al., 2023, the Quadratic Unconstrained Binary Optimisation (QUBO) formulation of the problem is defined in two parts: the core function H^0, which encodes the objective function of the problem, and the penalty function H^R, which is higher for unfeasible solutions. To facilitate the translation of the QUBO formulation to the Ising model used in QAOA, JRP will be formulated as a cost minimisation problem rather than a gain maximisation problem. H^0 will be negated, while H^R will penalise by increasing the cost value.

The core function is defined in Eq. (2). Each binary variable x_{ij} represents whether worker j is reassigned to vacant job i. It can be noted that the value of H^0 decreases for higher priority and affinity gains.

$$H^0 = -\sum_{ij} S_{ij} x_{ij} = -c^{\mathcal{P}} \sum_{ij} \left(\Delta^{\mathcal{P}}_{ij} x_{ij}\right) - c^{\mathcal{A}} \sum_{ij} \left(\Delta^{\mathcal{A}}_{ij} x_{ij}\right). \tag{2}$$

On the other hand, Eq. (3) defines the total penalty function, which is the sum of two terms. Its first term is presented in Eq. (4). This ensures, for a sufficiently large $\lambda^R_1 > 0$, that each vacant job i can be taken by at most one worker. Its second term is presented in Eq. (5). It ensures, for a sufficiently large $\lambda^R_2 > 0$, that each worker j can be reassigned to at most one vacant job.

$$H^R = H^R_1 + H^R_2. \tag{3}$$

$$H^R_1 = \lambda^R_1 \sum_i \left(\sum_j x_{ij} - 0.5\right)^2. \tag{4}$$

$$H^R_2 = \lambda^R_2 \sum_j \left(\sum_i x_{ij} - 0.5\right)^2. \tag{5}$$

The constants λ^R_1 and λ^R_2 are the **penalty coefficients** and indicate the degree of penalisation in case the equivalent constraints are not satisfied. The value must be configured heuristically by the user, and while a small value may not penalise sufficiently, an excessively large value may bias the objective function when using an optimiser.

Combining Eqs. (2), (4) and (5), the QUBO formulation of JRP can be defined as indicated in Eq. (6).

$$\begin{aligned} H = &-\sum_{ij} \left[c^{\mathcal{P}} \left(\mathcal{P}^V_i - \mathcal{P}^C_j\right) + c^{\mathcal{A}} \left(\mathcal{A}^V_{ij} - \mathcal{A}^C_{jj}\right)\right] x_{ij} \\ &+ \lambda^R_1 \sum_i \left(\sum_j x_{ij} - 0.5\right)^2 + \lambda^R_2 \sum_j \left(\sum_i x_{ij} - 0.5\right)^2. \end{aligned} \tag{6}$$

3 The benchmarking process

3.1 The Four-layer Design

The benchmarking process follows a four-layer design. A sample is generated given a set of configurations, where each individual in the sample corresponds to a set of JRP instances implemented in QAOA under a specific configuration. The proposed design is illustrated in Fig. 1, while the details of each layer are provided in Appendix A.1.

Each layer introduces an additional level of abstraction to the procedures involved in the benchmarking process. The **configuration sampling** layer is responsible for generating each individual in the output sample, ensuring that each one corresponds to a distinct input configuration. Each individual can be considered a sub-sample itself, as it encompasses the complete analysis of QAOA for five JRP instances. The generation of individuals for these sub-samples is managed by the **instance sampling** layer, where each individual represents the resolution of one of the five JRP instances. The structure of these layers is illustrated in the appendix at Fig. 8 and Fig. 9.

The third layer of abstraction, named the **algorithmic performance analyser**, is responsible for solving a JRP instance, processing the results and computing the algorithmic performance metrics. These metrics, along with other implementation data, are then stored in a subsample and used for further analysis. The layer design is illustrated in Fig. 10. Finally, the **QAOA solver** and the **brute-force solver** are introduced. The former implements the core aspects of QAOA, including its configuration and execution, while the latter employs a conventional brute-force approach to identify the optimal solution to the problem. The design of the QAOA Solver is shown in Fig. 11.

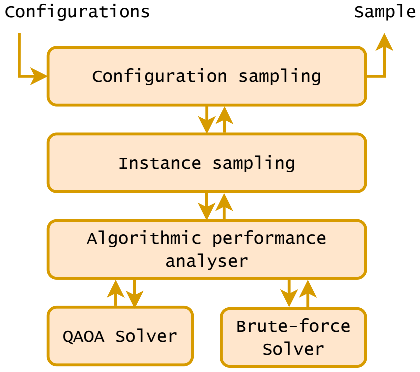


Fig. 1. 4-layer design for benchmarking QAOA on JRP.

3.2 Procedures Setup

A total of 21 configurations are defined as the input for the benchmarking process. Most of the hyperparameters are fixed to ensure a reliable sample size, considering hardware limitations, while only a subset of the hyperparameters has been selected for benchmarking.

All configurations use gain coefficients set to 1, ensuring equal importance is assigned to both affinity and job priority. The penalty coefficients are configured as $\lambda_1^R = 2.5$ and $\lambda_2^R = 3$. These values were determined based on informal tests conducted with small 4-qubit and 6-qubit instances. While it is expected that these values will perform well with larger instances, it cannot be formally proven that this will hold true.

Regarding the QAOA setup, a Trotterised Quantum Annealing (TQA) initialisation for variational parameters is used (Sack & Serbyn, 2021). The standard X-mixer is configured as the Mixer Hamiltonian. In each optimisation step, 10000 ansatz measurements are taken to calculate the expectation value, while only 20 measurements are performed during the evaluation step. This evaluation step determines the number of candidate solutions that will be found by QAOA after the optimisation. It is important to note that these candidates must pass the best candidate filter, as outlined in Fig. 11, in order to identify the best solution. The Powell optimiser (Powell, 1964) is used, with a maximum of 10000 iterations and a tolerance of 0.01. All of these configurations were implemented using the OpenQAOA SDK (Sharma et al., 2022).

Finally, the configuration parameters explored during the benchmarking process are defined. These include the number of workers, the number of vacant jobs, and the QAOA hyperparameter p. In Table 1, these distinct parameters are presented across 21 configurations, each identified by a tuple (workers, vacantJobs, p). Given the number of configurations, with each producing a subsample of five JRP instances analysed under QAOA, the total number of JRP instances analysed in the benchmarking process is $21 \times 5 = 105$.

ID	workers	vacantJobs	p
(5,3,3)	5	3	3
(5,3,4)	5	3	4
(5,3,5)	5	3	5
(3,5,3)	3	5	3
(3,5,4)	3	5	4
(3,5,5)	3	5	5
(6,3,3)	6	3	3
(6,3,4)	6	3	4
(6,3,5)	6	3	5
(3,6,3)	3	6	3
(3,6,4)	3	6	4
(3,6,5)	3	6	5
(5,4,3)	5	4	3
(5,4,4)	5	4	4
(5,4,5)	5	4	5
(4,5,3)	4	5	3
(4,5,4)	4	5	4
(4,5,5)	4	5	5
(4,4,3)	4	4	3
(4,4,4)	4	4	4
(4,4,5)	4	4	5

Table 1. Tables with the 21 input configurations for the benchmarking. Only the benchmarked parameters are outlined here, while the prefixed ones are mentioned in Section 3.2.

The selection of circuit depths $p \in \{3, 4, 5\}$ for the benchmarking is justified by established findings regarding algorithm performance and hardware capabilities. Crucially, layerwise QAOA training is known to saturate at a depth $p^* = n$, meaning that further increasing depth beyond the number of qubits n yields no additional improvement in overlap with the target state (Campos et al., 2021). A JRP instance has a circuit width of $workers \times vacantJobs = n$, resulting in these saturation points to be around $p^* \in \{15, 16, 18, 20\}$ for the established configuration parameters. Therefore, choosing depths $p \in \{3, 4, 5\}$ allows for the observation of significant performance improvements without encountering the saturation limitations. Furthermore, these depths are directly supported by existing benchmarkings. Numerical optimizations for QAOA have been conducted for systems up to $N = 20$ qubits, confirming the relevance of studying performance at these scales (Niu et al., 2019). Most notably, a recent comprehensive cross-platform QPU benchmarking study using the Linear Ramp QAOA (LR-QAOA) protocol included 56-qubit and 28-qubit instances with depths as low as $p = 3$, considering them informative and relevant for evaluating QPU coherence and comparing different hardware vendors (Montanez-Barrera et al., 2025).

3.3 Algorithmic Performance Metrics

The first metric is the approximation ratio, a 0-to-1 value that expresses how approximate is a solution obtained with respect to the optimal one. The closer the value is to 1, the better the approximation. This has been widely used in research on approximate algorithms for optimisation problems (Choi & Kim, 2019). It will help to quantify the quality of the obtained solutions.

Then, the Ising difference is defined in Eq. (7) as the difference between the ansatz expectation value evaluated with optimal variational parameters $\langle H \rangle_{\theta_{\text{opt}}}$ and the one evaluated with TQA-initialised variational parameters $\langle H \rangle_{\theta_{\text{TQA}}}$. Since it is not normalised through absolute value, the difference is given with a sign. In that way, given an Ising difference of $\pm a$, the larger the value of a, the bigger the difference between the evaluated Ising costs, while the sign $-$ or $+$ indicates whether there is a decreasing or increasing difference. Ising differences shall result in values with negative sign in order to show that the algorithm is theoretically working well.

$$\text{Ising difference} = \pm a = \langle H \rangle_{\theta_{\text{opt}}} - \langle H \rangle_{\theta_{\text{TQA}}} \tag{7}$$

The last defined metric is the gain improvement, which will allow to quantify how do QAOA's results improve the current productivity of an industry or organisation when it comes to solve JRP. It is defined in Eq. (8) as a percentage improvement between the gain of the JRP equation in its standard formulation (as a maximization problem with constraints) evaluated with a given solution G_{QAOA} and the gain with the solution that applies no reassignments G_{baseline}. While it may seem to provide the same information as the approximation ratio, it does not. Finding highly approximate solutions to an optimisation problem does not necessarily lead to a significant increase in an organisation's productivity. In some cases, applying the method to solve the problem may not be worthwhile, even if it achieves approximate solutions.

$$\text{Gain improvement} = \frac{G_{\text{QAOA}} - G_{\text{baseline}}}{G_{\text{baseline}}} \times 100\% \tag{8}$$

4 Result and Discussion

In Fig. 2, the approximation ratios for the 105 JRP instances solved by QAOA are presented. Each category along the x-axis corresponds to one of the 21 configurations that were set up. Additionally, the approximation ratios are grouped by problem size and hyperparameter p in Fig. 3. The distributions are represented using box plots, where each box is defined by the mean (indicated by the yellow line), the first and third quartiles (represented by the edges of the box), and the minimum and maximum values (depicted by the whiskers). The white points represent outliers.

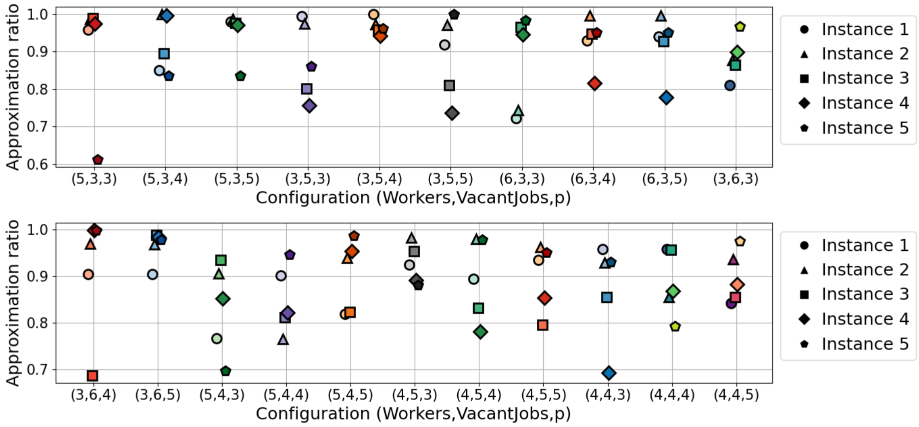


Fig. 2. Approximation ratios for the 105 JRP instances optimised with QAOA.

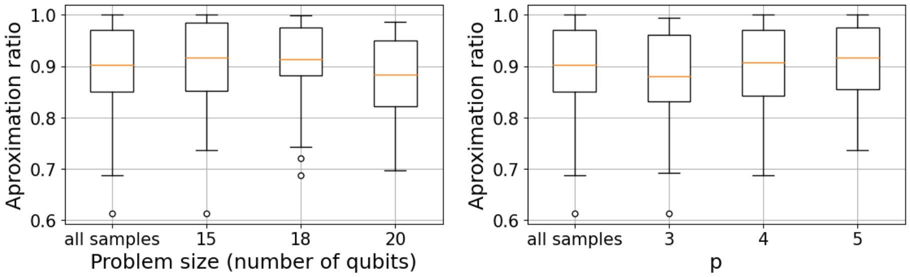


Fig. 3. The distributions of approximation ratios grouped by problem size on the left and by hyperparameter p on the right.

It can be easily seen from the plots that **QAOA generally performs quite well** in approximating the optimal JRP solutions, as most of the solutions found for the instances show approximation ratios in the range $[0.86, 0.97]$, with a mean value of approximately 0.9. Upon further examination of the grouped distributions, some underlying patterns emerge. The growth in problem size appears to correlate with a general decrease in the approximation ratio, which is to be expected given that the search space expands exponentially with problem size.

In contrast, an increase in the hyperparameter p seems to correspond to a general improvement in the approximation ratio, aligning with the theoretically anticipated behaviour and providing further **evidence of the algorithm proper functioning**. Moreover, increasing the p value appears to reduce the variance in the approximation ratios. This suggests that selecting an adequately large p value may not only improve the approximation quality of the JRP solutions but also **enhance the algorithm robustness** across multiple samplings for candidate solutions. For example, in a real-world scenario, the same problem instance may repeat over a long period of time, and the user may need to rerun new samplings for candidate solutions with months or even years between them.

In Fig. 4, all the Ising differences are plotted, with the results grouped by problem size and hyperparameter p in Fig. 5. These findings are **consistent with the previous results**, as the Ising differences are all in high negative values, i.e. negative values far from 0. This also shows that the algorithm is effectively minimizing the expectation value of the prepared ansatz.

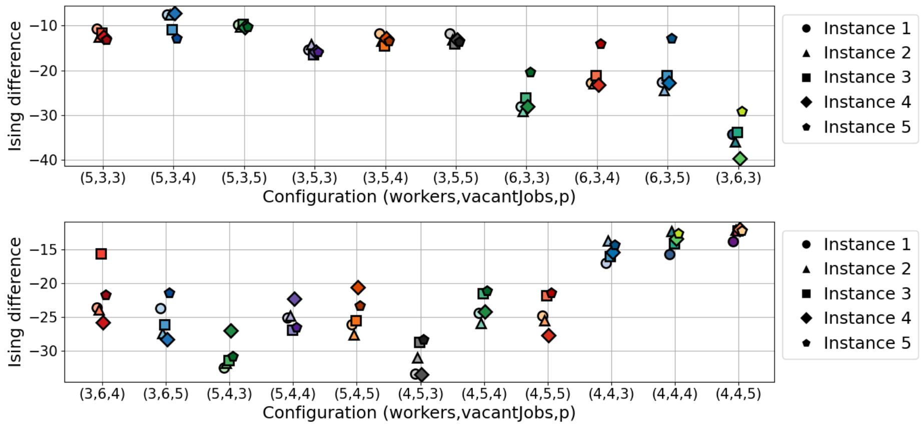


Fig. 4. Ising differences for the 105 JRP instances optimised with QAOA.

When analysing the distributions grouped by problem size, it is observed that the Ising differences consistently increase in the negative direction with the increase of problem size. In a first hypothesis, it might be assumed that this is related to the number of possible reassignments across different instances. Larger instances are likely to have more vacant jobs, providing more potential reassignments that reduce the Ising cost of both the optimal solution and its

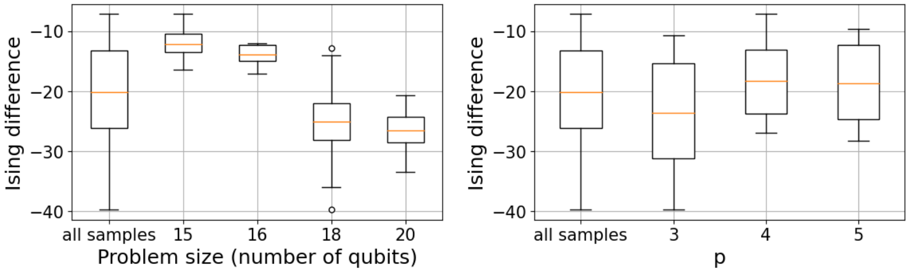


Fig. 5. The distributions of Ising differences grouped by problem size on the left and by hyperparameter p on the right.

approximate solutions. If this is the case, it should **not** be interpreted as **a significant factor** influencing the ability of QAOA to find more approximate solutions for larger instances. Instead, it reflects the tendency of larger instances to have optimal solutions with a lower Ising cost, making them farther from the initial Ising cost.

A second hypothesis states that the exponential growth of the search space as the problem size increases would cause the TQA initialisation to begin with worse variational parameters, i.e., starting farther from the optimal ones. This, in turn, would cause the Ising difference to increase in the negative direction and likely result in **longer optimisation time**.

Both the first and second hypothesises mentioned could occur simultaneously. In any case, a more detailed comparison of the Ising costs contributing to the Ising differences would be necessary to evaluate these hypotheses.

Another important factor that can affect the success of the method is the variability in the Ising differences across different problem sizes. It could be seen that the distribution for the 18-qubit instances is more spread out compared to the 16-qubit distribution. This could be related to the configuration of the number of workers and vacant jobs. For the 18-qubit instances, two configurations were set: one with 6 workers and 3 jobs (named $(6, 3, p)$) and the other as the inverse (named $(3, 6, p)$). For the 16-qubit instances, just one configuration with 4 workers and 4 jobs was set. Switching between configurations $(6, 3, p)$ and $(3, 6, p)$ is anticipated to lead to different Ising differences, and these cannot be generalised solely based on problem size. If that is the case, it will impact the approximation ratios achieved in instances, in a way that instances of one configuration could result in higher approximation ratios than the other. This underscores the **importance of a comparison per configuration** when evaluating JRP performance in QAOA, as merely considering the problem size may lead to inaccurate conclusions, particularly as the number of workers and vacant jobs becomes more imbalanced.

Regarding p, it was expected that the Ising difference would increase in the negative direction as p increases, with the hope that this would indicate that QAOA was finding better solutions. However, the observed trend was the opposite. Upon further analysis, an initial hypothesis attributes this behaviour to the method used to initialise variational parameters. As mentioned above, a TQA initialisation was employed. Previous work by Sack and Serbyn, 2021 has mentioned that decomposing the Quantum Annealing algorithm using a TQA approach over a discrete set of evolution times t_i with $(i = 1, \ldots, p)$ results in a unitary circuit equivalent to the depth-p QAOA ansatz. This mapping between TQA and QAOA, along with the universality of Quantum Annealing for $T \to \infty$, provides theoretical support for using TQA initialisation in QAOA. Based in the former, in an idealised scenario where $p \to \infty$, QAOA's ansatz would require no optimisation at all because the TQA initialization would correspond to the optimal parameters. However, in practical implementations where p is finite, the further p is from this idealised limit, the farther the initialization will be from the optimal parameters. This explains the observed trend where for larger p values, which approach the idealised infinite-depth regime, the Ising differences are nearer to 0. In other words, the result of the current analysis highlights the **strength of TQA-based initialisation in combination with higher p values**, not only in improving the quality of the solution but also in initializing parameters closer to the optimal ones.

Finally, Fig. 6 provides an overview of all observed gain improvements, while Fig. 7 illustrates how these varies with problem size. Taking into account the industrial context, the QAOA application in JRP yields a general gain improvement ranging between 9% and 15%, with an average improvement of 12%. Also, as the problem size increases, the gain improvement exhibits a decreasing trend. This observation aligns with previous analyses comparing problem size with respect to approximation ratios and Ising differences, further reinforcing the impact of problem complexity on QAOA performance. Either way, these main results show the reliability in solving JRP with QAOA and the considerable **improvement with respect to the current job assignments setting**.

5 Conclusion and Future Work

We applied QAOA to the Job Reassignment Problem (JRP), evaluating its performance on 105 small instances in a noiseless simulation. The analysis showed that QAOA effectively approximates optimal JRP solutions, achieving a mean approximation ratio near 0.9, with most solutions in the range of $[0.86, 0.97]$. As expected, the approximation ratio decreased with larger problem sizes. However, higher values of the QAOA hyperparameter p improved solution quality and robustness. From an industrial perspective, QAOA resulted in job reassignment efficiency improvements of 9% to 15%, with an average of 12%, demonstrating its potential for enhancing job allocation strategies.

The study also examined the Ising differences, showing that QAOA consistently minimises energy expectations as supposed. Interestingly, the variability in Ising differences was found to depend not only on problem size but also on the specific configurations of the number of workers and the number of vacant jobs. This suggests that evaluating JRP performance in QAOA requires more

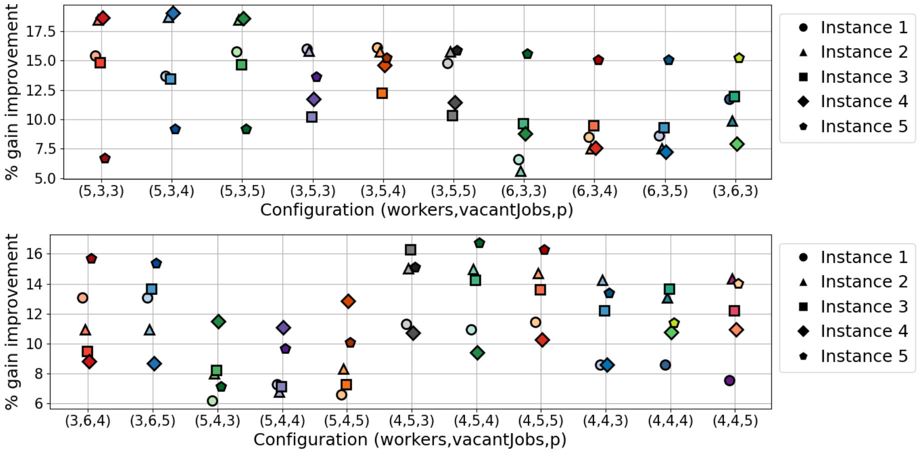


Fig. 6. Percentage gain improvement for the 105 JRP instances optimised with QAOA.

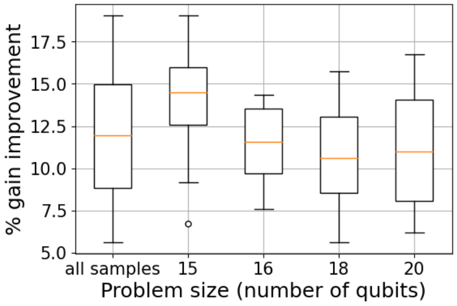


Fig. 7. The distributions of percentage gain improvements grouped by problem size.

than just analysing problem size. It is also needed a finer-grained comparison of the imbalance between the number of workers and number of vacant jobs.

Moreover, we observed that larger p values resulted in lower Ising differences, which initially appeared counter-intuitive. However, this trend aligns with prior work on TQA initialisation, supporting the idea that for sufficiently high p, QAOA approaches an idealized regime where the initialized variational parameters accurately approximate the optimal ones. These findings highlight the potential benefits of leveraging TQA-inspired initialisation in combination with large p values.

There are two main directions for future work. The first involves leveraging advanced statistical techniques and data mining to extract further relevant performance insights from the current data. This can includes attempting to extrapolate performance trends for larger problem instances and estimating the minimum number of measurements required to ensure a baseline solution quality.

The second direction focuses on exploring hardware support to actually test larger instances of the problem, ideally on real quantum devices. This would provide the opportunity not only to evaluate more realistic problem instances but also to assess the impact of device noise on the results. Additionally, experimenting with various error mitigation and suppression techniques would be valuable to identify the most effective approaches in this context. In doing so, it could be gained insight into the robustness of solutions in the presence of real-world quantum hardware noise.

References

Abbas, A., Ambainis, A., Augustino, B., Bärtschi, A., Buhrman, H., Coffrin, C., Cortiana, G., Dunjko, V., Egger, D. J., Elmegreen, B. G., Franco, N., Fratini, F., Fuller, B., Gacon, J., Gonciulea, C., Gribling, S., Gupta, S., Hadfield, S., Heese, R., . . . Zoufal, C. (2024). Challenges and opportunities in quantum optimization. *Nature Reviews Physics*, *6*(12), 718–735. https://doi.org/10.1038/s42254-024-00770-9

Campos, E., Rabinovich, D., Akshay, V., & Biamonte, J. (2021). Training saturation in layerwise quantum approximate optimization. *Physical Review A*, *104*(3), L030401.

Choi, J., & Kim, J. (2019). A tutorial on quantum approximate optimization algorithm (qaoa): Fundamentals and applications. *2019 international conference on information and communication technology convergence (ICTC)*, 138–142.

Cohen, D., Cooper, M., Jeavons, P., & Krokhin, A. (2004). A maximal tractable class of soft constraints. *Journal of Artificial Intelligence Research*, *22*(1), 1–22.

De Luca, G. (2022). A survey of nisq era hybrid quantum-classical machine learning research. *Journal of Artificial Intelligence and Technology*, *2*(1), 9–15.

Delgado, I. P., Markaida, B. G., Ali, A. M., & de Leceta, A. M. F. (2023). Qubo resolution of the job reassignment problem. *2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC)*, 4847–4852.

Farhi, E., Goldstone, J., & Gutmann, S. (2014). A quantum approximate optimization algorithm. https://arxiv.org/abs/1411.4028

Gemeinhardt, F., Garmendia, A., Wimmer, M., Weder, B., & Leymann, F. (2023). Quantum combinatorial optimization in the nisq era: A systematic mapping study. *ACM Computing Surveys*, *56*(3), 1–36.

Hadfield, S., Wang, Z., Rieffel, E. G., O'Gorman, B., Venturelli, D., & Biswas, R. (2017). Quantum approximate optimization with hard and soft constraints. *Proceedings of the Second International Workshop on Post Moores Era Supercomputing*, 15–21.

Hmer, A., & Mouhoub, M. (2010). Teaching assignment problem solver. *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, 298–307.

Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval research logistics quarterly*, *2*(1-2), 83–97.

Meseguer, P., Rossi, F., & Schiex, T. (2006). Soft constraints. In *Foundations of artificial intelligence* (pp. 281–328, Vol. 2). Elsevier.

Montanez-Barrera, J., Michielsen, K., & Neira, D. E. B. (2025). Evaluating the performance of quantum process units at large width and depth. *arXiv preprint arXiv:2502.06471*.

Niu, M. Y., Lu, S., & Chuang, I. L. (2019). Optimizing qaoa: Success probability and runtime dependence on circuit depth. *arXiv preprint arXiv:1905.12134*.

Papadimitriou, C. H., & Steiglitz, K. (1998). *Combinatorial optimization: Algorithms and complexity*. Courier Corporation.

Pentico, D. W. (2007). Assignment problems: A golden anniversary survey. *European Journal of Operational Research*, *176*(2), 774–793.

Powell, M. J. D. (1964). An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal*, *7*(2), 155–162. https://doi.org/10.1093/comjnl/7.2.155

Rajak, A., Suzuki, S., Dutta, A., & Chakrabarti, B. K. (2022). Quantum annealing: An overview. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, *381*(2241). https://doi.org/10.1098/rsta.2021.0417

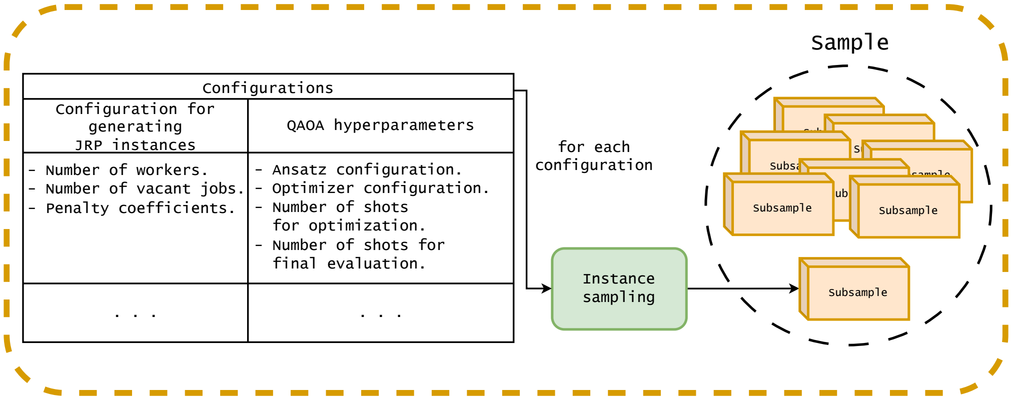
Sack, S. H., & Serbyn, M. (2021). Quantum annealing initialization of the quantum approximate optimization algorithm. *quantum*, *5*, 491.

Sharma, V., Saharan, N. S. B., Chiew, S.-H., Chiacchio, E. I. R., Disilvestro, L., Demarie, T. F., & Munro, E. (2022). Openqaoa–an sdk for qaoa. *arXiv preprint arXiv:2210.08695*.

Tilly, J., Chen, H., Cao, S., Picozzi, D., Setia, K., Li, Y., Grant, E., Wossnig, L., Rungger, I., Booth, G. H., & Tennyson, J. (2022). The variational quantum eigensolver: A review of methods and best practices. *Physics Reports*, *986*, 1–128. https://doi.org/10.1016/j.physrep.2022.08.003

A Appendix

A.1 Layers design



Configurations	
Configuration for generating JRP instances	QAOA hyperparameters
- Number of workers. - Number of vacant jobs. - Penalty coefficients.	- Ansatz configuration. - Optimizer configuration. - Number of shots for optimization. - Number of shots for final evaluation.
. . .	. . .

Fig. 8. Configuration sampling, first component of the 4-layer design.

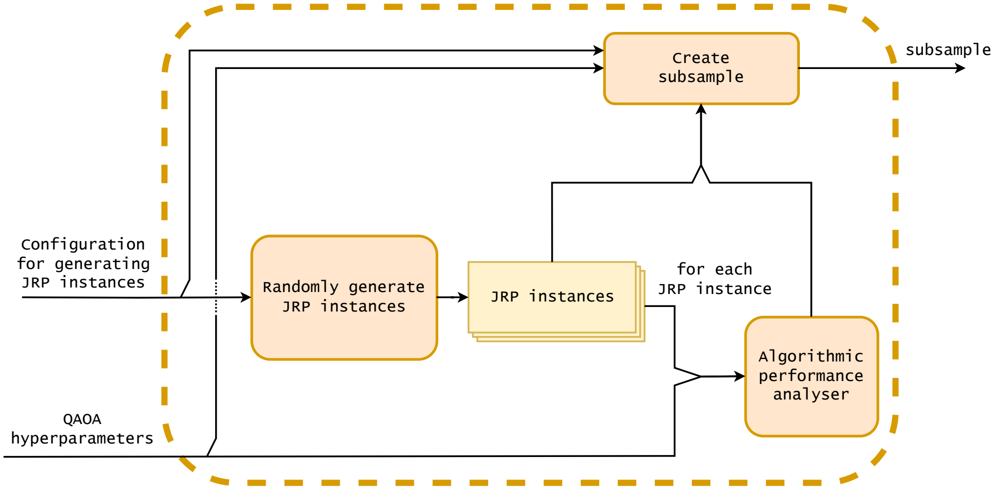

Create subsample
subsample
Configuration for generating JRP instances
Randomly generate JRP instances
JRP instances
for each JRP instance
Algorithmic performance analyser
QAOA hyperparameters

Fig. 9. Instance sampling, second component of the 4-layer design.

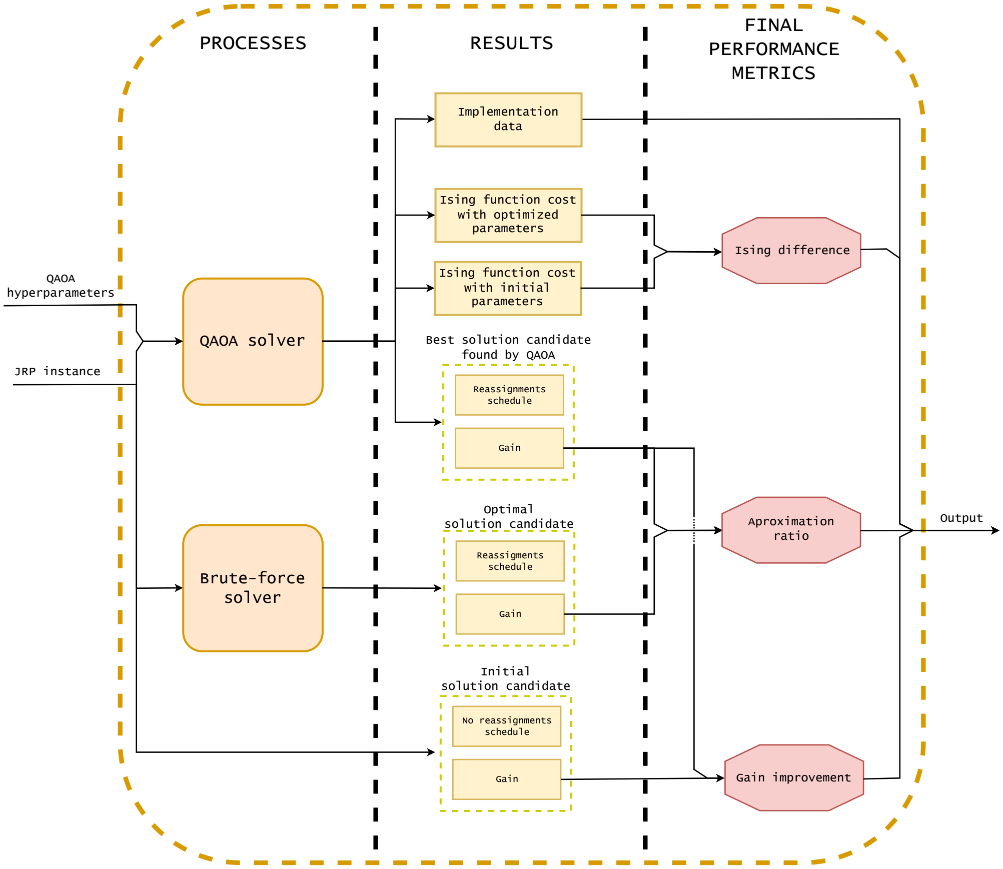

PROCESSES
RESULTS
FINAL PERFORMANCE METRICS
QAOA hyperparameters
JRP instance
QAOA solver
Brute-force solver
Implementation data
Ising function cost with optimized parameters
Ising function cost with initial parameters
Best solution candidate found by QAOA
Reassignments schedule
Gain
Optimal solution candidate
Reassignments schedule
Gain
Initial solution candidate
No reassignments schedule
Gain
Ising difference
Aproximation ratio
Gain improvement
Output

Fig. 10. Algorithmic performance analyser, third component of the 4-layer design.

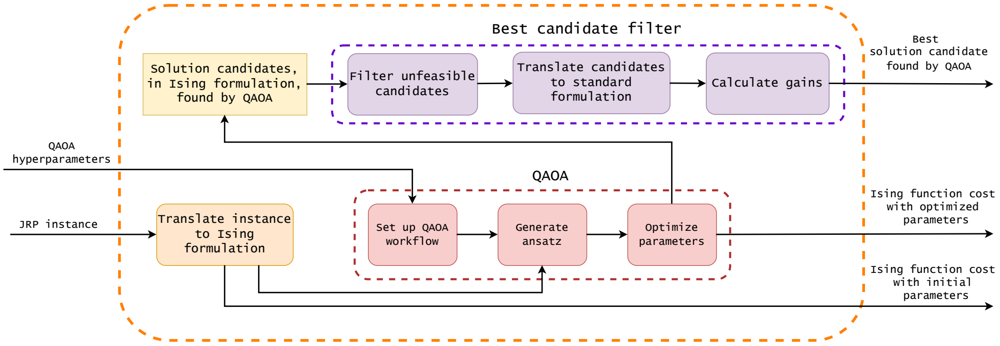

Best candidate filter
Solution candidates, in Ising formulation, found by QAOA
Filter unfeasible candidates
Translate candidates to standard formulation
Calculate gains
Best solution candidate found by QAOA
QAOA hyperparameters
JRP instance
Translate instance to Ising formulation
QAOA
Set up QAOA workflow
Generate ansatz
Optimize parameters
Ising function cost with optimized parameters
Ising function cost with initial parameters

Fig. 11. QAOA solver, the component which implements the core aspects of QAOA, including its configuration and execution.