Poznan University of Technology
Faculty of Computing and Telecommunications
Institute of Computing Science

Doctoral dissertation

APPLICATION OF QUANTUM COMPUTING APPROACHES FOR SOLVING OPTIMIZATION PROBLEMS

Tomasz Pecyna

Supervisor
Rafał Różycki, Ph. D., Dr. Habil, University Professor

Auxiliary tutor
Krzysztof Kurowski, Ph. D., Dr. Habil

Poznań, 2024

The doctoral dissertation was carried out as part of the "Doktorat Wdrożeniowy" program, funded by the State budget and conducted in collaboration with the Poznań University of Technology and Poznan Supercomputing and Networking Center. Agreement no. DWD/4/24/2022.

Abstract

Quantum computing leverages phenomena such as superposition, entanglement, and tunneling to perform computations on quantum processors. Several quantum algorithms, including Grover's algorithm and the Quantum Approximate Optimization Algorithm (QAOA), have been developed for solving optimization problems. However, limited research exists on the practical performance of these algorithms when applied to real-world problems. Additionally, the community has not yet developed high-level tools that facilitate both experimentation and the deployment of quantum solutions to real-life applications. This doctoral thesis addresses the challenge of applying quantum algorithms to different optimization problems. It provides detailed analyses of well-known quantum algorithms such as QAOA, the Quantum Alternating Operator Ansatz, the Feedback-based Algorithm for Quantum Optimization, and a novel approach involving the relaxation of constraints. These methods are applied for solving selected classical optimization problems like the Job Shop Scheduling Problem, as well as real-world optimization problems, such as Tactical Aircraft Deconfliction and the Electric Motor Vehicle Charging Problem. Furthermore, several high-level tools designed for users interested in quantum applications are introduced. These include a platform for launching combinatorial optimization problems, the modular tool QCG-QuantumLauncher for solving problems with quantum algorithms on real quantum hardware and a new decision support tool for human air traffic controllers . The thesis presents a cohesive, self-propelling loop, where research and application tools work symbiotically to advance the field of quantum computing for solving optimization problems.

List of publications

The dissertation consists of the introductory section and the following six original publications:

[P1] Różycki, R., Józefowska, J., Kurowski, K., Lemański, T., Pecyna, T., Subocz, M., Waligóra, G.: A quantum approach to the problem of charging electric cars on a motorway. Energies. 16, 442 (2022). https://doi.org/10.3390/en16010442.

Ministry points / journal: 140

[P2] Kurowski, K., Pecyna, T., Slysz, M., Różycki, R., Waligóra, G., Węglarz, J.: Application of quantum approximate optimization algorithm to job shop scheduling problem. European Journal of Operational Research. 310, 518–528 (2023). https://doi.org/10.1016/j.ejor.2023.03.013.

Ministry points / journal: 140

[P3] Pecyna, T., Kurowski, K., Rózycki, R., Waligóra, G., Węglarz, J.: Quantum variational algorithms for the aircraft deconfliction problem. In: Lecture notes in computer science. pp. 307–320 (2024). https://doi.org/10.1007/978-3-031-63778-0_22.

Ministry points / conference: 140

[P4] Pecyna, T., Różycki, R.: Improving quantum optimization algorithms by constraint relaxation. Applied Sciences. 14, 8099 (2024). https://doi.org/10.3390/app14188099.

Ministry points / journal: 100

[P5] Pecyna, T., Siera, D., Bosak, B.: QCG-QuantumLauncher: a modular tool for quantum scenarios. Conference materials from 15th International Conference on Parallel Processing & Applied Mathematics, Ostrava, Czech Republic, September 8-11, 2024

Ministry points / conference: 20

Contents

Chapter 1

Introduction

Quantum computing is a field of science that studies the use of quantum phenomena to perform computations. These phenomena include superposition, entanglement, and quantum tunneling. A device capable of performing such computations in a universal sense, i.e., one that implements a Quantum Turing Machine [Benioff, 1980], is called a quantum computer. The basic unit of information in a quantum computer is a qubit, and the basic building blocks of operations are quantum gates. In addition to universal quantum computers, there are many other quantum devices that do not fully implement the Quantum Turing Machine but are used to execute specific types of computations. These include, for example, quantum annealers and quantum bosonic samplers.

Optimization problems involve identifying the best subset of solutions based on chosen evaluation criteria, while also ensuring that these solutions meet a specified set of constraints, making them feasible. In this context, a solution refers to any assignment of values to decision variables. Optimization problems can be classified into different types based on various factors. One classification separates them based on the type of variables used, which can be continuous or discrete. A problem is considered mixed if it involves both continuous and discrete variables. Optimization problems can also be categorized by the nature of the solution set, which can be either finite or infinite. Optimization problems, along with decision, search, counting, and function problems, form a class of computational problems that are studied within the field of computational complexity, which focuses on the efficiency of algorithms. Many optimization problems are hard to solve with existing computers due to their inherent computational complexity.

Quantum computing is a relatively new field of science compared to classical computing. The first quantum computers with only a few qubits were developed in the late 1990s [Chuang et al., 1998]. Building quantum computers is a difficult task due to various engineering challenges connected to quantum mechanics. Consequently, the applications of quantum computing to solving optimization problems are mostly limited to theoretical work and proof-of-concept implementations on Noisy Intermediate-Scale Quantum (NISQ) computers [Preskill, 2018]. These implementations aim to demonstrate the potential of quantum approaches, despite the challenges posed by noise and decoherence in current quantum devices. By nature, quantum algorithms are probabilistic, meaning they do not guarantee finding exact solutions. Instead, they return solutions with certain probabilities, depending on the number of final measurement samples. For a high-level perspective and information about quantum computing, one can refer to PSNC's Quantum Report `https://quantum.psnc.pl/wp-content/uploads/2023/11/Raport_Quantum_ENG-1.pdf`.

Undoubtedly, one of the two most famous quantum algorithms is Grover's algorithm [Grover, 1996]. This algorithm is one of the first to reveal the potential quantum advantage in solving real-life problems. Grover's algorithm is capable of finding an element in an unstructured

database in $\mathcal{O}(\sqrt{N})$ function evaluations, which constitutes a quadratic speedup compared to the best classical algorithm available [Bennett et al., 1997]. Due to its quantum advantage and versatility, Grover's algorithm is also used as a subroutine in more complex algorithms. For example, it can be used to address optimization problems by combining Quantum Minimum Finding with dynamic programming [Ambainis et al., 2019].

Even though current quantum computers can offer over a hundred qubits, which could theoretically demonstrate a quantum advantage over classical computers if the qubits were ideal, in practice, their capabilities are limited to successfully running computations only on a few dozen qubits and several dozen quantum layers [Lubinski et al., 2023]. This is far less than what is required to successfully run Grover's algorithm, let alone its extensions. Algorithms that seem to cope much better with NISQ computers are so-called variational algorithms, with two prominent examples being the Variational Quantum Eigensolver (VQE) [Peruzzo et al., 2014] and the Quantum Approximate Optimization Algorithm (QAOA) [Farhi et al., 2014]. Both of these are hybrid classical-quantum algorithms, meaning that the quantum operations are supported by classical computations.

Due to NISQ era limitations, current research and applications of quantum computing to optimization problems can be categorized into three different groups:

- Application research dedicated mainly to deepening the understanding of popular algorithms solving problems such as MaxCut [Zhou et al., 2020]. These have only narrow applications, e.g., in the silicon industry.

- Applications focused on reapplying already established rules for problem formulation specific to non-universal quantum devices, such as quantum annealers, to a broad range of problems, for example traffic flow optimization or portfolio optimization [Yarkoni et al., 2022].

- Application research that transfers formulations developed for non-universal quantum devices directly into universal paradigms for selected real-world problems, without conducting an in-depth analysis or fully exploiting the potential of quantum computers, for example by solving the tail assignment problem [Vikstål et al., 2020].

The number of papers on the considered topic that do not fall into these three groups is very limited.

On the other hand, a healthy ecosystem encompassing elements like scientific research, industry application, newcomer training, software development, and hardware integration, all of which are well-functioning, creates optimal conditions for a wide range of long-term advancements in the area. As for quantum computing, the community is actively working on scientific research [Abbas et al., 2023], and gate-level programming tools are well developed [Javadi-Abhari et al., 2024]. However, there is a substantial need for industry application and high-level abstractions, particularly those integrated with high-performance computing [Schulz et al., 2022], to enable users to focus on applications rather than learning and implementing technical details.

The following thesis addresses the aforementioned shortcomings in the applications of universal quantum computing paradigms and the lack of high-level abstractions. The topic of this dissertation, application of quantum computing approaches for solving optimization problems, is fulfilled in a multifaceted manner. Firstly, the focus is on solving a well-known scheduling problem with high applicability in real-life scenarios. Additionally, formulations for less-known problems are introduced and solved predominantly using the QAOA and its variants on both emulators of quantum computers and real hardware. For comparison, some results are benchmarked against

those obtained from quantum annealers. This work is not limited to developing formulations to be solved by quantum algorithms but also proposes a novel approach for addressing such problems. The applications are ultimately delivered in the form of two specialized, yet user-friendly platforms, both of which utilize a modular tool called QCG-QuantumLauncher, a modular tool developed for solving quantum scenarios. Specifically, the added values of the doctoral dissertation include the contributions listed below.

- Using the time-indexed representation of the Job Shop Scheduling Problem (JSSP) developed by [Venturelli et al., 2015] and inspired by the educated-guess strategy demonstrated for the MaxCut problem [Zhou et al., 2020], the Hamiltonian for both the decision and optimization versions of the JSSP was formulated. Patterns in the variational parameters of QAOA were analyzed, and the results were discussed, highlighting the relationship between measured energy and achieved makespans of schedules. Additionally, different instances and the time required for optimization using the proposed approach were examined. The early results were presented during the 18th International Workshop on Project Management and Scheduling, and the full results were published in a scientific journal (**P2**).

- Research on solving the JSSP using the QAOA was conducted. A new platform based on QCG[1] [Piontek et al., 2016] was developed, enabling non-experienced users to launch well-known optimization problems, such as Exact Cover, MaxCut, and JSSP, on various quantum computers and simulators. That platform development contributed to a software tool for managing and solving multiple combinatorial optimization problems on various quantum devices was developed. Named QCG-QuantumLauncher, the tool was further enhanced and evolved into a modular platform for quantum scenarios, which was subsequently presented at a scientific conference (**P5**).

- Addressing the potential future problem of electric vehicle charging due to their increased numbers, a quantum approach to schedule and optimize their target charging locations on a motorway was proposed. A collision matrix was introduced as an intermediate step in formulating the problem Hamiltonian. The QAOA approach launched on a quantum simulator was compared with D-Wave quantum annealer, and the results were published in a scientific journal (**P1**).

- Building on the lessons learned from solving the aforementioned problem, the tactical deconfliction problem was investigated as part of the Quantum Air Traffic Management project. The problem was formulated, quantum circuits were launched on both quantum simulators and real quantum hardware, and different instances were compared, contrasting the standard QAOA with the Quantum Alternating Operator Ansatz algorithm [Hadfield et al., 2019]. The early results were presented during the 19th International Workshop on Project Management and Scheduling, and the full results were presented at a scientific conference (**P3**).

- Further expanding research on the Quantum Air Traffic Management project and the tactical deconfliction problem, a novel approach for formulating quantum Hamiltonians for variational algorithms by relaxing constraints was proposed. Experiments were conducted on instances with varying levels of difficulty, and the results were published in a scientific paper (**P4**).

- All research on the tactical deconfliction problem (**P3**, **P4**) contributed to the successful completion of the Quantum Air Traffic Management project. As part of this project, a

[1] `https://qcg.psnc.pl/`

supporting tool for human flight operators to perform aircraft deconfliction was developed. This tool, named SkyDodge, can use gate-based quantum hardware, quantum annealers, or photonic devices to perform the deconfliction.

The rest of the paper covers those contributions in greater details. Chapter 2 establishes the necessary foundations of quantum computing. In Chapter 3, all the research work is described and results are presented. The applications of the work are discussed in Chapter 4. Finally, Chapter 5 summarizes the work.

Chapter 2

Quantum computing

2.1 Quantum computations

In quantum computing, the basic unit of information is a qubit. Physically, a qubit can be, for example, an ion in an electromagnetic trap or a photon. Mathematically, a qubit is a two-level state that can be represented as a vector in a Hilbert space. Typically, to express qubits, we choose a computational basis, and use Dirac's bra-ket notation: $|0\rangle = \left(\begin{smallmatrix}1\\0\end{smallmatrix}\right)$ and $|1\rangle = \left(\begin{smallmatrix}0\\1\end{smallmatrix}\right)$. One property that differentiates qubits from classical bits is their ability to be in a superposition of two basic states. For a qubit $|\psi\rangle$, we can write it as:

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle\,, \tag{2.1}$$

where α denotes the amplitude that the qubit has with respect to the state $|0\rangle$, and β denotes the amplitude that the qubit has with respect to the state $|1\rangle$. Amplitudes are physical values that can be represented as elements from the set of complex numbers. They correspond to the probability of a qubit collapsing to the corresponding state upon measurement, which is an action aimed at observing and therefore determining a qubit's state. For example, measuring a qubit that is physically a photon can be achieved by passing it through a polarizing filter. Mathematically, the probability of a state $|\psi\rangle$ collapsing into a state $|m\rangle$ after measurement is given by:

$$p(m) = \langle\psi|\, M_m^\dagger M_m\,|\psi\rangle\,, \tag{2.2}$$

where M_m is an element from a set of measurement operators, satisfying the completeness relation $\sum_m M_m^\dagger M_m = I$, which ensures that the probabilities sum up to one, i.e., $\sum_m p(m) = 1$. For the qubit $|\psi\rangle$, taking $M_0 = |0\rangle\langle 0|$ and $M_1 = |1\rangle\langle 1|$ and performing simple calculations, we achieve:

$$p(0) = \alpha^2,\ p(1) = \beta^2,\ \alpha^2 + \beta^2 = 1. \tag{2.3}$$

Having a physical qubit, for example, an ion in an electromagnetic field, we can change its state by applying a laser on it with a given frequency. Mathematically, this is equivalent to performing a unitary evolution:

$$|\psi'\rangle = U\,|\psi\rangle\,, \tag{2.4}$$

where U is a unitary linear operator. Unitary linear operators that we apply to qubits are called quantum gates. For example, to transform a qubit in the $|0\rangle$ state to an equal superposition, we can employ the Hadamard gate, $\mathrm{H} = \frac{1}{\sqrt{2}}\left(\begin{smallmatrix}1 & 1\\ 1 & -1\end{smallmatrix}\right)$, obtaining $\mathrm{H}\,|0\rangle = \frac{1}{\sqrt{2}}\,|0\rangle + \frac{1}{\sqrt{2}}\,|1\rangle = |+\rangle$. In a

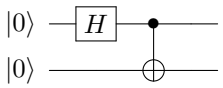


FIGURE 2.1: Circuit definition of a Bell state $|\Phi^+\rangle$.

similar fashion, we can deal with a set of qubits, called a quantum register. Having n qubits, we can write $\mathrm{H}^{\otimes n}|0\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}}\sum_{x\in\{0,1\}^n}|x\rangle = |+\rangle^{\otimes n}$. Additionally, we often write $|xy\rangle = |x\rangle \otimes |y\rangle$ as a shortcut for the tensor product.

The most distinctive property of qubits is the possibility to entangle them. Succinctly speaking, entanglement constitutes a situation where two or more qubits cannot be considered and described independently of each other. Moreover, by performing an operation on one qubit, we automatically affect the other entangled qubits. The simplest entangled states are the Bell states, for example, $|\Phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$. To create entanglement gates, we usually use the two-qubit Controlled-NOT gate (also called CX), $\mathrm{CNOT} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes X$, where $X = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$. Starting with two qubits in the $|00\rangle$ state, the $|\Phi^+\rangle$ state can be achieved as follows:

$$|\Phi^+\rangle = \mathrm{CX}(\mathrm{IH}|00\rangle) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle). \tag{2.5}$$

However, it is more common to use quantum circuits as a description language for the evolution of quantum states. The circuit evolution of Equation 2.5 is shown in Figure 2.1. There are many more quantum gates and specific states that are frequently used. For a comprehensive list, the reader is referred to well-known textbooks, e.g., [Nielsen and Chuang, 2001]. In general, any unitary operator U can be written in the form of an evolution:

$$U = \exp(iH), \tag{2.6}$$

for a Hermitian operator H. Moreover, any Hermitian operator with dimension $n^2,\ n \in \mathbb{N}$ can be constructed from the sum of Pauli matrices with real coefficients:

$$X = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right),\ Y = \left(\begin{smallmatrix} 0 & -i \\ i & 0 \end{smallmatrix}\right),\ Z = \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right),\ I = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right). \tag{2.7}$$

This is especially useful when constructing parameterized rotation operators. When we consider the rotation of a qubit, we usually think about rotation around a specified axis of a qubit represented as a Bloch sphere [Feynman et al., 1957]. For example, a θ-parameterized rotation around the x-axis is defined by:

$$RX(\theta) = e^{-i\frac{\theta}{2}X} = \begin{pmatrix} \cos\frac{\theta}{2} & -i\sin\frac{\theta}{2} \\ -i\sin\frac{\theta}{2} & \cos\frac{\theta}{2} \end{pmatrix}. \tag{2.8}$$

Similar ideas apply for creating many-qubit rotation gates. For example, a 2-qubit rotation about the ZZ axis has the following mathematical representation:

$$R_{ZZ}(\theta) = e^{-i\frac{\theta}{2}ZZ} = \begin{pmatrix} e^{-i\frac{\theta}{2}} & 0 & 0 & 0 \\ 0 & e^{i\frac{\theta}{2}} & 0 & 0 \\ 0 & 0 & e^{i\frac{\theta}{2}} & 0 \\ 0 & 0 & 0 & e^{-i\frac{\theta}{2}} \end{pmatrix}. \tag{2.9}$$

Note that the R_{ZZ} gate is an entangling gate that splits the phases of the states $|00\rangle$ and $|11\rangle$ from the states $|01\rangle$ and $|10\rangle$.

Sometimes, when we deal with n qubits, we write $Z_{i_1} \dots Z_{i_m}$, where $i_1 < \dots < i_m,\ m \leq n$, to represent a term:

$$Z_{i_1} \dots Z_{i_m} = \underbrace{I \dots I Z\, I \dots I Z}_{i_m - 1} \underbrace{I \dots I}_{i_1 - 1}. \tag{2.10}$$

For example, for a quantum register of 5 qubits, $Z_1 Z_4 = IZIIZ$.

For the purpose of this dissertation, two more theorems that are strongly interconnected need to be mentioned. Firstly, quantum gates are universal, meaning there exist finite sets of quantum gates that can be used to approximate any quantum operation to arbitrary accuracy. This implies that a small, fixed set of quantum gates is sufficient to construct any quantum algorithm, ensuring the feasibility of implementing complex quantum operations through sequences of these elementary gates. Examples of such sets include parameterized rotation gates together with the phase shift gate and CNOT gate, or the set comprising the Clifford set and the T gate [Nielsen and Chuang, 2001]. However, the construction of some unitary operations might be costly in some cases, meaning that an exponential number of basic gates might be needed. Fortunately, there is the Solovay–Kitaev theorem [Kitaev, 1997], which states that we can approximate any arbitrary one-qubit gate up to ϵ with only $\mathcal{O}(\log^c \frac{1}{\epsilon})$, where c is a small constant, using gates from one of the universal sets. For an algorithm containing m CNOTs and unitary gates, the circuit can be approximated with $\mathcal{O}(m \log^c \frac{m}{\epsilon})$, which is believed to be a sufficient complexity for most applications [Nielsen and Chuang, 2001]. These two theorems build the theoretical foundations for the practicality of building real quantum computers, as most quantum hardware operates on some discrete set of single-qubit gates and a limited number of double-qubit gates, which result in a universal set.

2.2 Quantum algorithms

There are several criteria by which we can classify existing quantum algorithms. One such criterion distinguishes quantum algorithms based on quantum hardware requirements. Algorithms that necessitate many (on the order of thousands) high-quality qubits with error correction are designed for so-called fault-tolerant quantum hardware. These include some of the most well-known algorithms, such as Grover's algorithm and Shor's algorithm, which offer theoretical guarantees of quantum speedup over classical algorithms. However, according to the most optimistic roadmaps of companies engineering quantum hardware, it will take at least a decade to reach the fault-tolerant era [IBM, 2023].

On the other side of the spectrum are algorithms applicable in the so-called Noisy-Intermediate Scale Quantum (NISQ) era. These algorithms cope much better with noisy environments and error-prone qubits that are likely to suffer from crosstalk or decoherence. The Quantum Approximate Optimization Algorithm (QAOA) and the Variational Quantum Eigensolver (VQE) are examples of such algorithms. These are hybrid variational algorithms, meaning they are essential by classical computations that optimize variational parameters, which are crucial for the functioning of these algorithms. However, this hybridity is not only needed for the NISQ era; algorithms are being developed that exhibit synergy from the integration of fault-tolerant quantum hardware and High Performance Computing (HPC). In fact, it is the classical computational power that allows for the early adoption of nascent quantum approaches. The remainder of this subsection will review the details of NISQ algorithms that are relevant to this dissertation.

2.2.1 Quantum Approximate Optimization Algorithm

The QAOA is inspired by and designed to mimic the behavior of the adiabatic theorem [Born and Fock, 1928]. This theorem states that a quantum system initially in an eigenstate will remain in that eigenstate if changes to the system are applied slowly enough. The idea is to prepare a state that is the ground state of a simple Hamiltonian and then slowly change this Hamiltonian to one that describes the solution to a given optimization problem. After measurement, we should then be able to observe a state that corresponds to a solution to our problem. Mathematically, we can describe it as

$$H(t) = \left(1 - \frac{t}{T}\right) H_M + \frac{t}{T} H_C, \tag{2.11}$$

where H_M is our initial Hamiltonian and H_C is the final Hamiltonian. The slow changes are performed by increasing t from $t = 0$ up to $t = T$. When considering optimization problems in QAOA algorithms and their variants, H_M is often referred to as the mixer Hamiltonian, while H_C is known as the cost (or problem) Hamiltonian. The two Hamiltonians in Equation 2.11 do not commute, so to perform Hermitian evolution (see Equation 2.6), the additional step of Trotterization must be taken. Trotterization is described by the following formula:

$$e^{A+B} = \lim_{p \to \infty} \left(e^{\frac{A}{p}} e^{\frac{B}{n}}\right)^p, \tag{2.12}$$

where p is an integer that serves as an approximation constant. Further expanding the idea by combining Equations 2.6 and 2.12, parametrizing a circuit with variational parameters controlled by an optimizer launched on a classical computer, and assuming the initial eigenstate to be the equal superposition of n qubits, we obtain the final QAOA formula:

$$|\psi_p(\overrightarrow{\gamma}, \overrightarrow{\beta})\rangle = e^{-i\beta_p H_M} e^{-i\gamma_p H_C} \cdots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_C} |+\rangle^{\otimes n}. \tag{2.13}$$

The $|+\rangle^{\otimes n}$ state is an eigenstate of the mixer Hamiltonian composed of Pauli-X gates, i.e., $H_M = \sum_{i=1}^{n} X_i$, and $\overrightarrow{\gamma}$ and $\overrightarrow{\beta}$ are sequences of variational parameters. These variational parameters are adjusted by a classical optimizer, which aims to minimize the expected value (also known as energy):

$$\min_{\overrightarrow{\gamma}, \overrightarrow{\beta}} \langle \psi_p(\overrightarrow{\gamma}, \overrightarrow{\beta}) | H_C | \psi_p(\overrightarrow{\gamma}, \overrightarrow{\beta}) \rangle. \tag{2.14}$$

This equation represents a specific type of measurement known as a projective measurement, as described in Equation 2.2.

For further insights about the vanilla QAOA, we refer the reader to the original paper [Farhi et al., 2014] and other detailed analyses [Zhou et al., 2020].

2.2.2 Quantum Alternating Operator Ansatz

Even though the connection between QAOA and the adiabatic theorem is not mathematically rigorous, it has proven beneficial to align initial states as eigenstates of the mixer Hamiltonian [He et al., 2023]. However, some approaches diverge from the adiabatic framework by exploring different interpretations. One such interpretation views the algorithm from a numerical perspective, noting that the cost Hamiltonian distinguishes preferred solutions in phase, while the mixer Hamiltonian is responsible for amplitude amplification and reduction. Another observation is that the mixer Hamiltonian facilitates transitions between different states. This insight is utilized in the Quantum Alternating Operator Ansatz (QAOAnsatz) algorithm, which aims to reduce the search

space by restricting exploration to solutions that satisfy hard constraints [Hadfield et al., 2019]. This is achieved by designing the mixer Hamiltonian to preserve two key properties. Firstly, it should preserve the feasible subspace, meaning transitions can only occur between states corresponding to feasible solutions. Secondly, it should allow for full exploration of the states corresponding to feasible solutions, meaning any pair of these states can be connected through unitary transformations derived from the Mixer Hamiltonian.

The general formula for the Quantum Alternating Operator Ansatz (QAOAnsatz) is similar to that of the standard QAOA, with one key modification: instead of starting the evolution from the equal superposition state, we begin from a state that corresponds to one of the feasible solutions to the problem (or a superposition of states corresponding to feasible solutions), denoted as $|\psi_0\rangle$:

$$|\psi_p(\overrightarrow{\gamma}, \overrightarrow{\beta})\rangle = e^{-i\beta_p H_M} e^{-i\gamma_p H_C} \cdots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_C} |\psi_0\rangle. \tag{2.15}$$

It is natural to consider states corresponding to feasible solutions when dealing with optimization problems. However, for decision problems, finding a feasible solution already constitutes a solution, and using the algorithm in such a case would not be meaningful. From an algorithmic perspective, the key objective is to maintain the relationship between the initial state and the mixer Hamiltonian, which ensures transitions within a subset of solutions and reduces the search space. Therefore, for decision problems, the mixer Hamiltonian can be designed to reduce the search space by restricting transitions between states to only a proper subset of the problem's constraints.

2.2.3 Feedback-based Algorithm for Quantum Optimization

Optimizing variational parameters can be challenging, especially for deeper QAOA/QAOAnsatz circuits with higher p. The method proposed by Magann et al. [Magann et al., 2022], called the Feedback-based ALgorithm for Quantum OptimizatioN (FALQON), addresses this optimization challenge by utilizing the circuit itself to determine the $\overrightarrow{\beta}$ parameter sequence while discarding the $\overrightarrow{\gamma}$ parameter sequence. This approach makes the algorithm purely quantum, eliminating the need for any classical optimization.

The algorithm starts by setting $\beta_1 = 0$ and applying a single evolution $e^{-i\beta_1 H_M} e^{-iH_C}$ to the equal superposition state $|+\rangle^n$. Subsequently, it measures the commutator observable between H_M and H_C:

$$A_k = -\langle\psi_k|i[H_M, H_C]|\psi_k\rangle. \tag{2.16}$$

This estimation is used to set $B_{k+1} = -wA_k$, where w serves as a hyperparameter. The process continues by incrementally deepening the circuit and estimating the next β_{k+1} until $k = p$, marking the end of the algorithm. Throughout this algorithm, H_C is also estimated to monitor convergence progress. Typically, when considering optimization problems, H_M and H_C are chosen to be the same as in the standard QAOA.

2.3 Quantum computational complexity

To manage expectations about quantum computers, it is important to introduce the concept of quantum computational complexity. This will help avoid the misconception that quantum computers can solve exponential, unstructured problems in polynomial time.

Scientists often view the **P** vs **NP** problem as a challenge to uncover a deeply hidden structure within **NP-Complete** problems, one that has yet to be discovered. Such a discovery could potentially lead to a proof that **P** = **NP**, or conversely, that **P** ≠ **NP**. Unfortunately, there

is little evidence supporting the idea that **P** might equal **NP**, and most scientists believe that **NP-Complete** problems do not have any intrinsic structure. If this is the case, we can consider a problem known to lack inherent structure, such as the unstructured database search, which is addressed by Grover's algorithm. This algorithm provides a quadratic speedup over classical algorithms, specifically solving the problem in $\pi\frac{\sqrt{N}}{4}$ steps [Grover, 1996], where **N** is the data size. Assuming $N = 2^n$, the algorithm requires around $2^{\frac{n}{2}}$ queries. While this is impressive and useful, it does not represent a breakthrough in terms of complexity classes. Moreover, this result is known to be optimal [Bennett et al., 1997], which suggests that quantum computers might also not be able to solve **NP-Complete** problems in polynomial time. It is important to remember that this statement is based on common-sense suggestions and opinions of the scientific community, rather than definite proofs.

There are, however, some positive indications of quantum advantage over classical computation from a computational complexity perspective. The hope is centered on the **BQP** class, which stands for Bounded-error Quantum Polynomial time. This class describes problems that can be solved by quantum circuits with a bounded probability of error in polynomial time. More specifically, it is defined as the class of languages accepted by a quantum Turing machine in polynomial time, with an error probability of at most $\frac{1}{3}$. It is known that **BQP** $\subseteq$ **BPP** (Bounded-error Probabilistic Polynomial time), and that **BQP** $\subseteq$ **PSPACE** (problems solvable by a Turing machine using polynomial space) [Bernstein and Vazirani, 1993], forming the following chain:

$$\mathbf{L} \subseteq \mathbf{P} \subseteq \mathbf{BPP} \subseteq \mathbf{BQP} \subseteq \mathbf{PSPACE} \subseteq \mathbf{EXP}. \tag{2.17}$$

Although there exist some algorithms which demonstrate strict inclusions between particular classes from the chain under a quantum oracle, such as the Deutsch–Jozsa algorithm [Deutsch and Jozsa, 1992], it remains unknown whether these inclusions are strict in general. However, it is straightforward to see that **P** $\subseteq$ **EXP** and **L** $\subseteq$ **PSPACE**. The motivation to work on quantum computing also arises from strong suggestions that **P** = **BPP**, as discussed for example in [Impagliazzo and Wigderson, 1997]. Moreover, the famous Shor's algorithm [Shor, 1994], which solves factoring problem in polynomial time, suggests that **BPP** might not equal **BQP**, which would formally reveal that computers are more powerful than classical ones for practical scenarios.

That being said, there is an expectation that the **BQP** class might solve some problems in **NP** that are not in **P** (assuming **P** $\neq$ **NP**). Moreover, there are indications that the **BQP** class might contain some problems that are outside **NP**, such as those in **PSPACE**.

Chapter 3

Research work

Referring back to the introduction, the goal of this dissertation is to contribute to the intersection of quantum computing, optimization problems, and their applications. Considering the current state of the art in research, along with the capabilities and limitations of quantum hardware and its future prospects, the most suitable algorithm to develop and explore new solutions and applications was the QAOA, along with its enhancements. At the time this work began, there was little research on the application of QAOA to significant areas. Therefore, it was a natural decision to pave the way for advancing this field by applying QAOA to complex, real-world optimization challenges and exploring its potential to push the boundaries of current quantum computing methods.

3.1 Formulating combinatorial problems as cost Hamiltonians

The main idea behind using the QAOA algorithm to solve optimization problems is to construct the problem Hamiltonian, which not only defines the quantum circuit but also serves as the cost function for the classical optimizer. A close analogy to classical optimization is representing the problem in terms of mathematical programming formulations, with objective criteria and variable constraints, specifically binary variables. Writing the problem in a binary mathematical programming form is also beneficial because the classical variables can be easily transformed to represent quantum quantities in a well-established spin Ising model [Lucas, 2014]. Each variable $x_1, \ldots, x_n$ can be replaced with the following linear operator:

$$x_i = \frac{I - Z_i}{2}. \tag{3.1}$$

This approach is typically used when creating analogues from classical quadratic binary optimization, as seen in quantum annealing, to gate-based quantum computers [Vikstål et al., 2020]. However, it is more common to directly define Hamiltonians or translate them from Boolean representations using derived formulas [Hadfield, 2021]. The construction of the problem Hamiltonian is particularly critical, as it also defines the circuit, as mentioned earlier. The different approaches to defining such Hamiltonians are also a subject of this dissertation, and the results of the differences between them will be discussed in the following sections.

One additional note to consider is that in classical mathematical optimization, the process involves an optimizer working to minimize or maximize the cost function while ensuring that the solutions remain within the feasible subspace, i.e., satisfying the constraints. In the standard QAOA algorithm, however, optimization involves minimizing the cost function while also minimizing the number of unsatisfied constraints. This approach effectively incorporates the optimization criteria directly into the cost function, which is similar to the Lagrange multiplier method [Bertsekas, 2014].

We will now briefly describe the application problems addressed in this dissertation and the base Hamiltonians used for these problems.

3.1.1 Job Shop Scheduling Problem (JSSP)

The Job Shop Scheduling Problem (JSSP) is one of the most well-known scheduling problems, with applications across various industries [Zhang et al., 2019]. The goal is to find the shortest possible schedule, known as the makespan, for a set of operations assigned to different machines. Operations within the same job must follow a predefined sequence. Additionally, the jobs are non-preemptive. The paper **P2** introduces a new Hamiltonian formulation for gate-based quantum computers, which had not been presented before. This formulation draws inspiration from previous works that approached JSSP using quantum annealing with a time-indexed representation [Venturelli et al., 2015]. For a complete derivation of the Hamiltonian formulas, we refer the reader to the original paper and present only the final outcome here.

Firstly, we need to constrain that each operation starts exactly once. The Hamiltonian responsible for this constraint is formulated as follows:

$$H_1^{JSSP} = \sum_k \left(I - \sum_t \frac{Z_{kt} - I}{2} \right)^2 , \tag{3.2}$$

where Z_{kt} corresponds to k-th operation starting at time t in the time-indexed representation.

Secondly, we need to ensure that only one operation is processed on a machine at any given time. Using the auxiliary set R_m (see, **P2**), this constraint is enforced by the Hamiltonian:

$$H_2^{JSSP} = \sum_m \left(\sum_{k,t,k',t' \in R_m} \frac{I - (Z_{kt} + Z_{k't'} - Z_{kt} Z_{k't'})}{4} \right) \tag{3.3}$$

Lastly, we need a Hamiltonian to ensure that the predefined order of operations, each of length l_k, within each of the J jobs is maintained. This is accomplished by the Hamiltonian:

$$H_3^{JSSP} = \sum_{i=1}^{J} \left(\sum_{\substack{k_{i-1}<k<k_i \\ t+l_k>t'}} \frac{I - (Z_{kt} + Z_{k+1,t'} - Z_{kt} Z_{k+1,t'})}{4} \right) \tag{3.4}$$

Summing these three Hamiltonians, we obtain a final Hamiltonian that fully represents the decision version of the JSSP:

$$H_D^{JSSP} = H_1^{JSSP} + H_2^{JSSP} + H_3^{JSSP} \tag{3.5}$$

For the optimization version of the JSSP, we need to introduce an additional Hamiltonian that adds a penalty based on the completion times, $t_{k_1} + l_1, t_{k_2} + l_2, \dots t_{k_n} + l_n$, of the last operations from each job, $k_1, k_2, \dots k_n$:

$$H_4^{JSSP} = \sum_{i=1}^{J} \frac{1}{2} \left(I - Z_{k_i, t_{k_i}} \right) (J+1)^{t_{k_i} + l_i} \tag{3.6}$$

Thus, for the optimization version of JSSP, the final Hamiltonian is:

$$H_O^{JSSP} = H_1^{JSSP} + H_2^{JSSP} + H_3^{JSSP} + H_4^{JSSP} \tag{3.7}$$

3.1.2 Electric Motor Vehicle Charging Problem (EMVCP)

In contrast to the JSSP, the problem of charging electric motor vehicles (EMVs) on a motorway does not have any standardized definitions. The paper **P1** simplifies the problem to a decision problem involving the scheduling of jobs on parallel, unrelated machines and explores multiple scenarios. For instance, it examines the differences between treating a single charger as a charging point versus treating an entire station as a charging point. While the paper explores both quantum annealing and QAOA approaches, the problem instances and definitions had to be simplified for the QAOA analysis.

In this paper, the Conflict Matrix (CM) was introduced for the first time. The Conflict Matrix is a lookup table designed to assist in scheduling Electric Motor Vehicles (EMVs) and predicting potential conflicts between them at a charging stations. It also serves as an auxiliary tool for constructing Hamiltonians, with the final formulations given below.

We need to ensure that each EMV charges at one and only one station. The Hamiltonian that enforces this constraint is formulated as follows:

$$H_1^{EMVCP} = \sum_{i=1}^{n} I - \sum_{j=1}^{s}\sum_{k=1}^{m_i}\left(\frac{1}{2}(I - Z_{ijk})\prod_{j'=1,j'\neq j}^{s}\prod_{k'=1,k'\neq k}^{m_i}\frac{1}{2}(I + Z_{ij'k'}))\right), \tag{3.8}$$

where Z_{ijk} corresponds to the ith EMV charging at station j and driving with mode k.

Additionally, we must ensure that the number of EMVs charging at a station at any given time does not exceed the station's capacity. The Hamiltonian that enforces this constraint is formulated as:

$$H_2^{EMVCP} = \sum_{p\subseteq s\in S:|p|=b_j+1} \frac{1}{2^{|p|}} \prod_{i'jk':x_{i'jk'}\in p} (I - Z_{i'jk'}), \tag{3.9}$$

where S is the set containing collections of EMVs which would potentially conflict with the ith EMV driving with mode k and charging at station j, for each such EMV, provided that the number of potential conflicting EMV's is greater than the stations's capacity b_j.

By summing these two equations, we obtain the final Hamiltonian for the problem:

$$H^{EMVCP} = H_1^{EMVCP} + H_2^{EMVCP}. \tag{3.10}$$

3.1.3 Tactical aircraft deconfliction problem

During the work on the Quantum Air Traffic Management (QATM) project led by Air Force Institute of Technology, and the consortium partner Poznan Supercomputing and Networking Center (PSNC)[1], it was proposed to address the tactical aircraft deconfliction problem using quantum approaches. This problem had not been previously tackled with quantum methods, with most classical approaches relying on mixed-integer linear and nonlinear programming techniques [Pelegrín and d'Ambrosio, 2022]. In a nutshell, the tactical deconfliction problem is about predicting and resolving aircraft conflicts in the airspace from 5 to 30 minutes ahead. A conflict is defined as a violation of a so-called safety cylinder of one aircraft by other aircraft.

The main idea of the developed approach was to propose, for each of the n aircraft, a set of m alternative maneuvers corresponding to changes in parameters such as flight speed, heading angle, or flight level. Subsequently, for each pair of these alternatives, it was determined whether they would conflict with each other. The results were recorded in a Conflict Matrix (CM), similar to the

[1] `https://itwl.pl/en/news/430-quantum-technologies-in-itwl`

one used in the EMVCP. To enable this approach for quantum optimization, two Hamiltonians need to be formulated.

Firstly, each aircraft must choose one and only one maneuver. This constraint can be enforced with the Hamiltonian:

$$H_1^{QATM} = \sum_{i=1}^{n} I - \sum_{j=1}^{m} \left(\frac{1}{2}(I - Z_{ij}) \prod_{j'=1, j' \neq j}^{m} \frac{1}{2}(I + Z_{ij'}) \right). \tag{3.11}$$

Secondly, if two maneuvers are chosen, there must no be a conflict between them, as is constrained by the Hamiltonian:

$$H_2^{QATM} = \sum_{i,j,i',j':\mathrm{CM}(i,j,i',j')=1} \frac{1}{4} I - \frac{1}{4}(Z_{ij} + Z_{i'j'} - Z_{ij} Z_{i'j'}). \tag{3.12}$$

When we combine the two Hamiltonians, we obtain the final Hamiltonian that can be used to correctly solve the tactical aircraft deconfliction problem:

$$H^{QATM} = H_1^{QATM} + H_2^{QATM}. \tag{3.13}$$

3.2 Performance of standard QAOA for solving optimization problems

The standard QAOA with default Hamiltonian formulations can be successfully used to solve combinatorial problems, as demonstrated in papers **P1** and **P3**. Both papers include performance analyses on quantum hardware simulators, which have limited capabilities due to the exponential costs of computational memory. However, as shown in **P3**, the results from the simulator do not differ significantly from those obtained using real quantum hardware provided by PSNC.

For both quantum hardware simulators and real quantum hardware, the sizes of solvable instances are currently limited to several dozen variables. For example, this corresponds to solving instances with 5 EMVs for the EMVCP, where there is a 0.5% probability of measuring the feasible solution on *ibmq_toronto*, as reported in **P1**. Similarly, for the tactical aircraft deconfliction problem, instances of size 5×4 can be solved with a probability of no more than 0.01% on *ibmq_torino*, as shown in **P3**. These instance sizes are significantly smaller compared to what can currently be addressed by specialized quantum annealers, which are capable of solving instances with up to 50 EMVs.

Both papers **P1** and **P3** highlight the importance of selecting the appropriate QAOA depth, as deeper circuits (those with larger p) more closely approximate adiabatic evolution and, in theory, might yield better results. However, on real quantum hardware and noisy simulators, deeper circuits are more susceptible to noise, decoherence, and crosstalk. The paper **P1** demonstrates that the best results are achieved with circuits of depth $p = 2$, as circuits with depth $p = 1$ do not approximate well, and circuits with depth $p = 3$ or more are unable to find feasible solutions. It is important to note, however, that the increase in circuit depth does not significantly affect the computation time for optimizing variational parameters; for example, the computation time for depth $p = 5$ is only 1.43 times greater than for depth $p = 1$.

The final takeaway from the experiments on standard QAOA is that careful attention must be paid to the formulation of the problem Hamiltonian, as the success probability can vary significantly with changes in the instance structure, even if the instance size remains the same. In particular, the experiments reported in **P3** for instances of size 12 variables showed that it is much easier to find feasible solutions when there are more aircraft and fewer maneuvers. Conversely, it is 2 to 3 times

harder to find feasible solutions when there are fewer aircraft and more maneuvers, even though the product of these two quantities remains the same as in the previous case. However, the switch from decision problems to optimization problems does not seem to introduce additional difficulty in solving the problem, as shown in **P2**. This is expected since the Hamiltonians for decision and optimization problems in this context differ only in one-qubit terms and do not introduce any additional entanglements. This same pattern can be applied to other optimization problems, such as the tactical aircraft deconfliction problem, as described in **P3**.

3.3 Hybrid classical-quantum QAOA approach

The close connection between QAOA and adiabatic evolution should also be evident in the behavior of variational parameters. Building on Equations 2.11 and 2.13, we can conjecture that the sequence of variational parameters $\overrightarrow{\gamma}$ should increase with circuit depth, while the sequence $\overrightarrow{\beta}$ should decrease with increasing depth. This pattern was first described by [Zhou et al., 2020], who proposed an approach that capitalizes on this property by searching for optimal variational parameters for shallow QAOA circuits, interpolating them, and then using them as initial seeds for optimization in deeper QAOA circuits. This approach was later verified and applied to the Tail Assignment Problem by [Vikstål et al., 2020].

This approach naturally suggests a synergy between quantum and classical hardware that can be exploited. The paper **P2** demonstrates the utilization of High Performance Computing (HPC) resources available at PSNC to find optimal variational parameters for the JSSP case. The paper first introduces a method to visualize optimal variational parameters using energy landscapes, illustrating that this property also applies to the JSSP. It then presents the results of such interpolations, starting from a QAOA depth of $p = 3$. For example, the paper reports up to a 6.5-fold decrease in optimization time between depths $p = 3$ and $p = 4$, provided that the optimal parameters found for a QAOA circuit of depth $p = 3$ are used as initial optimization points for depth $p = 4$.

A related issue worth mentioning is the connection between energy levels and the quality of solution when dealing with optimization problems. Just as in classical artificial intelligence, where a cost function is minimized or maximized, energy is minimized in quantum optimization, as derived in Section 2.2.1. However, when the probability distribution is considered, understanding this relation can be challenging for those focused on practical applications. Therefore, visualizing this connection, as demonstrated in **P2**, can be beneficial for a clearer understanding. The results show that, in some cases, infeasible solutions may have lower energy than feasible solutions, particularly in makespan optimization. Nonetheless, the general rule that lower energy corresponds to a higher probability of obtaining high-quality solutions remains valid. Moreover, the analysis highlights the potential for tuning decision parameters, such as the maximum time T for JSSP instances, to improve results.

3.4 Pure quantum optimization

The compute resources and overall summed computation time required for the approach described in Section 3.3 are significantly high, reaching over 180 CPU days of simulated QPU to find optimal variational parameters for some instances. On the other end of the spectrum, there is the idea of removing the classical part entirely, with one such approach being the FALQON algorithm mentioned in Section 2.2.3. Although never published, experiments on the FALQON algorithm

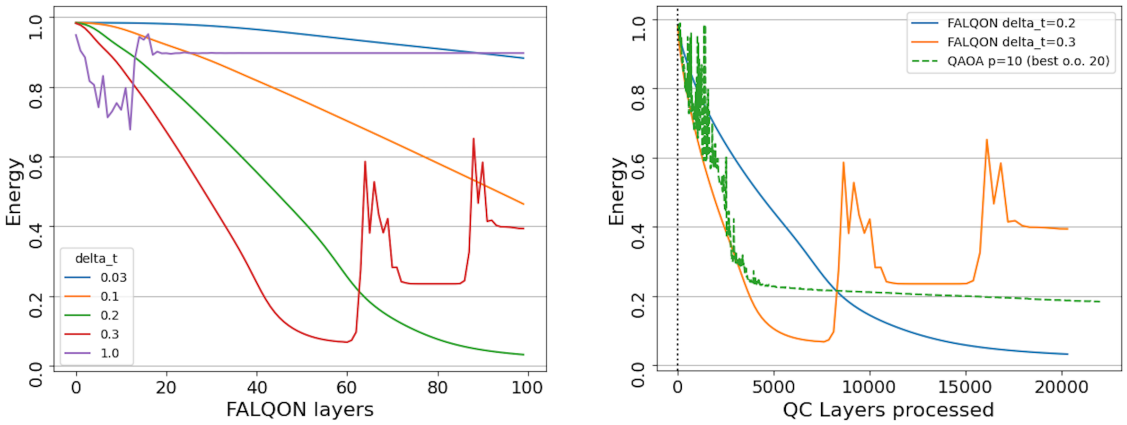


FIGURE 3.1: Exemplary energy results for the FALQON algorithm solving Exact Cover instances as a function of cumulative processed circuit layers. Left: comparison of different values of the parameter δ_t. Right: comparison of two optimization processes with the best δ_t parameters against standard QAOA with depth $p = 10$.

have been described in various presentations and seminars. A summary of these experiments which are relevant to this dissertation is presented in Figure 3.1.

The results indicate that the FALQON algorithm can achieve significantly lower energy levels compared to standard QAOA. However, it is important to note that while the number of QAOA layers remains constant throughout the algorithm, FALQON requires iterative circuit deepening, which increases the probability of noise affecting the outcomes. The results shown in Figure 3.1 were obtained using ideal simulators, meaning they should be considered as a potential roadmap for future optimization rather than a practical solution on current NISQ hardware.

3.5 Reducing the search space

Another degree of freedom that might be explored to further improve results in quantum optimization is the approach to integrate hard constraints into the mixer Hamiltonian using the QAOAnsatz algorithm, as described in Section 2.2.2. This approach reduces the search space, increasing the probability of measuring a feasible solution, albeit at the cost of adding more entangling gates to the circuit. Not all hard constraints need to be integrated into the mixer Hamiltonian, as it may not be beneficial for every constraint. For decision problems, including all constraints is generally not applicable. However, for the reference problems discussed in Sections 3.1.1, 3.1.2, and 3.1.3, we can extract a subset of constraints to incorporate into the mixer Hamiltonian, and this approach has proven to be quite effective.

For the Tactical Aircraft Deconfliction problem, embedding the hard constraint that each aircraft performs exactly one maneuver into the mixer Hamiltonian increased the probability of measuring a feasible solution in the RCP instance with 5 aircraft, each with 3 alternative maneuvers, from 11.69% to 77.33%, representing an over sixfold increase, as shown in **P3**. However, this approach did not significantly improve the overall shape of the probability distribution in the optimization aimed at minimizing the total number of changes required to the original flight schedule. Mathematically, it is enought to replace the standard mixer Hamiltonian with:

$$H_M = \sum_{i=1}^{n} X_{im}X_{i1} + Y_{im}Y_{i1} + \sum_{j=1}^{m} X_{ij}X_{ij+1} + Y_{ij}Y_{ij+1}. \tag{3.14}$$

In other problems, such as the JSSP and EMVCP, the one-hot constraint can be incorporated in the same way as it was for the Tactical Aircraft Deconfliction problem. Alternatively, other

problem-specific constraints may require different approaches, such as controlled operations, which are described in more detail in [Hadfield et al., 2019].

3.6 Relaxation of problem formulations

At first glance, it may seem counterintuitive, but taking the opposite approach to the one described in Section 3.5 yields similar results. Specifically, relaxing the problem formulation to consider a broader range of solutions as feasible, while simplifying the number of entanglements, also appears to deliver better outcomes compared to the standard QAOA approach.

The entirely novel approach, described for the first time in **P4** relaxes the entanglement-heavy constraint that forces aircraft to perform one and only one maneuver, by allowing aircraft to perform more maneuvers. More precisely, the Hamiltonian from Equation 3.11 is replaced by the Hamiltonian representing the NOT XOR function:

$$H^{QATM}_{\text{NOT XOR}} = -\tfrac{1}{2}I + \tfrac{1}{2}Z_1Z_2 \dots Z_m. \tag{3.15}$$

.

With this modification, a performance improvement is evident not only in comparison to standard Hamiltonian formulations but also to QUBO formulations that include at most second-degree entanglements. For instance, in a scenario involving 3 aircraft and 5 maneuvers, the paper reports a probability of measuring a feasible solution at 0.0009%, compared to 0.0007% for the QUBO approach and 0.0001% for the standard formulation, within a particularly challenging setup where only 5 feasible solutions exist out of 2^{15}. These results strongly suggest that universal quantum computation may become more preferable in the future than quantum annealers.

Chapter 4

Application Results

Except for the most basic concepts and a few well-known algorithms, quantum computing is generally considered challenging to learn, particularly for those without a strong theoretical background in mathematics, physics, or computer science. This is especially relevant in the context of industry applications, where engineers aiming to optimize their processes may not have the time or inclination to delve deeply into quantum computing to evaluate its applicability to their specific use cases. This chapter of the dissertation focuses on the tools developed during the four-year PhD studies period, including both applications created for the internal use of the Poznan Supercomputing and Networking Center and products developed for external clients, as well as outcomes of public projects. All the developed solutions are collected and accessible on PSNC's quantum website, `https://quantum.psnc.pl/en/`

4.1 PSNC Hybrid Classical-Quantum Platform for Optimization Algorithms

As a consortium member, PSNC participated in the EuroHPC PL project[1] led by Academic Computer Center CYFRONET AGH, which aimed to build national infrastructure for large-scale computing for research and industry. PSNC, as the consortium partner, was responsible for developing a platform for quantum operational research and discrete optimization, as well as acquiring access to and purchasing infrastructure for quantum combinatorial optimization. The former, with a Technology Readiness Level (TRL) [Héder, 2017] of 9, was a central focus of the applied research in this dissertation. The project's objectives were met by extending and integrating the existing QCG software stack [Piontek et al., 2016], and where necessary, developing new components.

During the development of this project, a web access layer was created to embed application templates and problem-oriented applications, offering a flexible and customizable platform for various use cases. This platform was built on the QCG-Portal component, enabling users to view, control, and monitor jobs through a web-based interface. It was further integrated with QCG-Templates, allowing for customized application views through parameterization. Additionally, QCG-API and QCG-Agent were employed for authorization and job submission. Figure 4.1 provides a high-level overview of the QCG software stack.

The developed platform includes several predefined graphical applications that allow users to define instances of selected and implemented combinatorial optimization problems, such as the MaxCut problem, Exact Cover problem, and the JSSP. The interface allows users to specify instance

[1] `https://www.eurohpc.pl/`

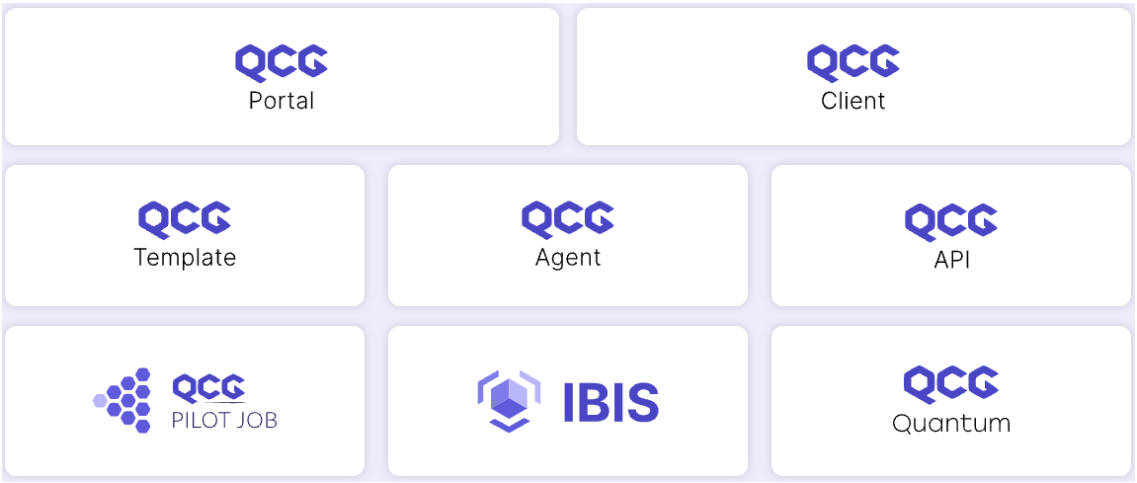


FIGURE 4.1: High-level view of the QCG software stack developed by PSNC (for more details, see `https://qcg.psnc.pl/qcg-building-blocks/`).

size and type using an intuitive graphical or text-based interface, or by loading pre-existing data from a file. Additionally, the platform supports the submission of generic computational tasks for any optimization problem by entering a QUBO matrix via the text interface or uploading it from a file. Users can then select the architecture on which the problem will be executed, including real quantum hardware or quantum simulators deployed during the PRACE-LAB2 project[2] led by PSNC. Multiple additional options allow users to choose the appropriate optimization algorithm and adjust its hyperparameters. It is also worth noting that the classical parts of the hybrid algorithms have been prepared for parallelization, enabling their execution on HPC clusters or high-performance computers, which significantly accelerates calculations. A screenshot showing the submission of a MaxCut instance is presented in Figure 4.2.

Applications for selected discrete optimization problems are accessible through the QCG-Portal, which has been adapted and functionally extended for the project. The QCG-Portal also serves as a tool for resource management, user authentication, task submission to the queuing system, and retrieval of metadata for submitted tasks. An exemple list of submitted tasks, along with their statuses and metadata, is shown in Figure 4.3. The developed templates within the platform also facilitate the analysis of task results via an interactive graphical interface, which presents results in the form of charts tailored to the specific problem, algorithm, and quantum architecture.

Note that the QCG components required additional enhancements to fully support quantum use cases, and views specific to these cases had to be implemented. One advanced tool that needed to be developed from scratch was the QCG-QuantumLauncher (QCG-QL). This tool serves as the backend of quantum algorithms enabling easy execution of combinatorial problems on quantum computers using quantum algorithms. The QCG-QL tool is further detailed in Section 4.3

4.2 Quantum-based decision support tool for air traffic control

The EuroHPC PL project was focused on providing users with a tool for solving selected combinatorial optimization problems or custom QUBO matrices on quantum machines. While these solutions were successfully implemented during the project, certain scenarios required more detailed and specific integration with existing processes. Such was the case in the Quantum Air Traffic Management (QATM) project[3] (TRL 6), where a quantum solution was developed to support air

[2] `https://prace-lab2.pl/`
[3] `https://itwl.pl/en/news/430-quantum-technologies-in-itwl`

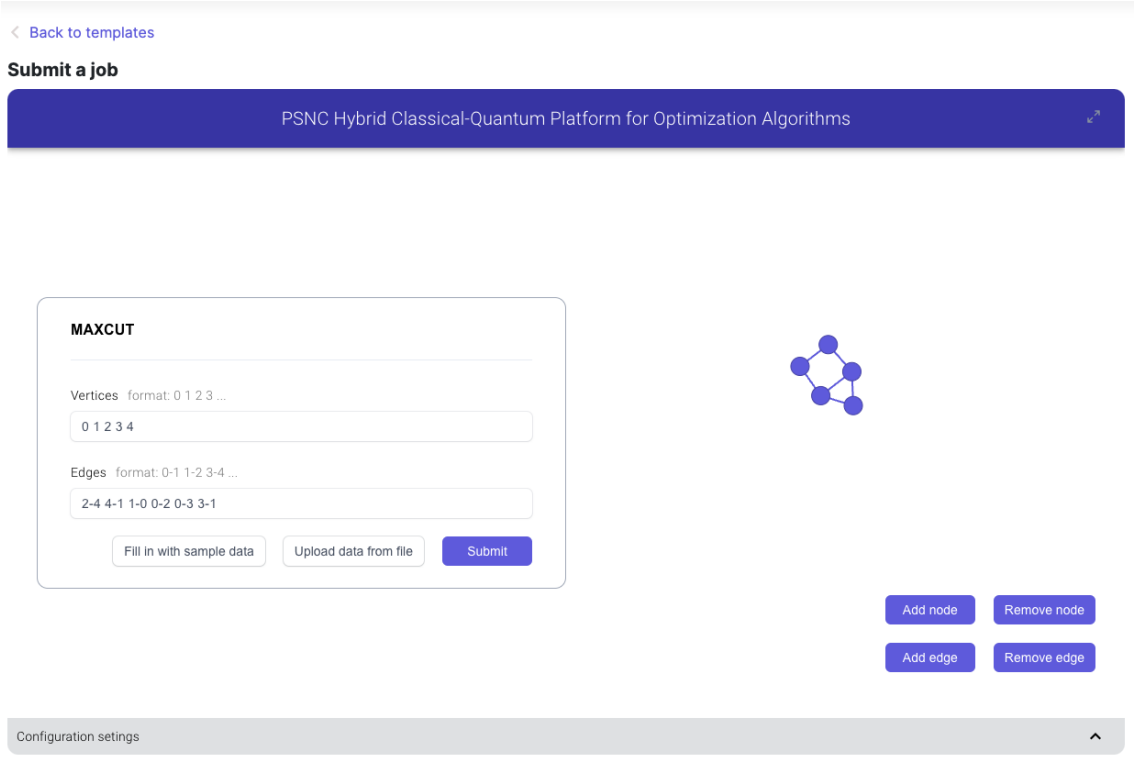


FIGURE 4.2: View of the QCG-Portal template for submitting the MaxCut problem for solving on a quantum computer.

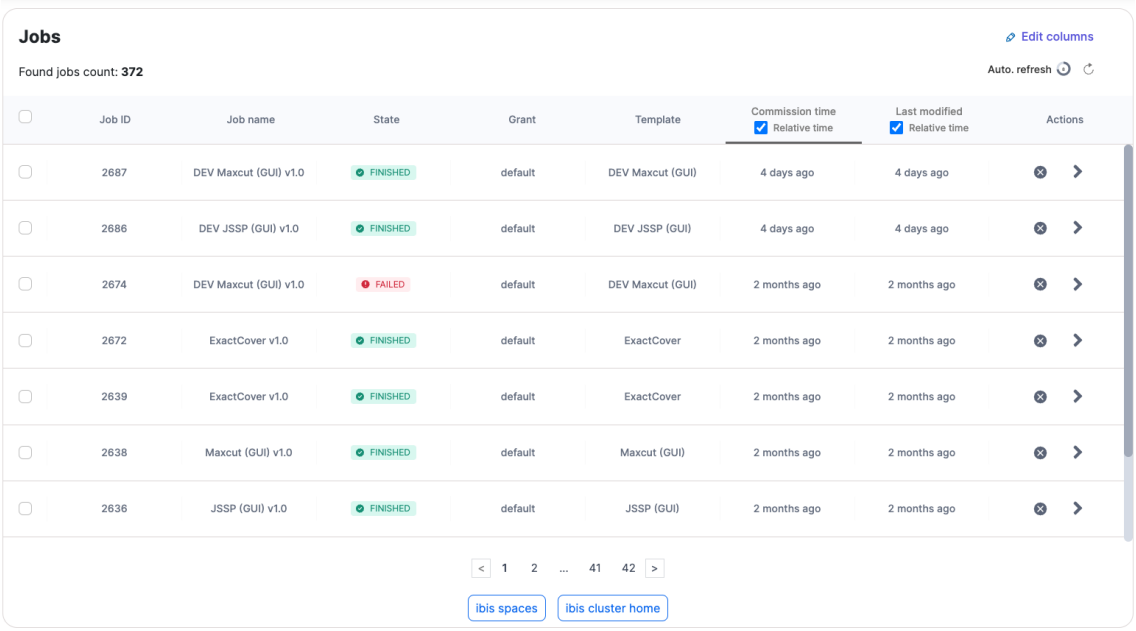


FIGURE 4.3: View of the QCG-Portal listing submitted jobs, their statuses and metadata.

traffic controllers. The gate-based component of this solution was addressed in the application part of this dissertation and is a natural continuation os the work described in Section 4.1.

The product developed during the project consists of three modules: the quantum backend, which processes input data about aircraft and outputs a solution to the problem, a QCG-Template accessible through the QCG-Portal, which enables launching computations on quantum hardware or classical simulators and SkyDodge, an interactive interface that monitors the airspace situation and displays potential solutions found by quantum machines. The quantum backend is based on

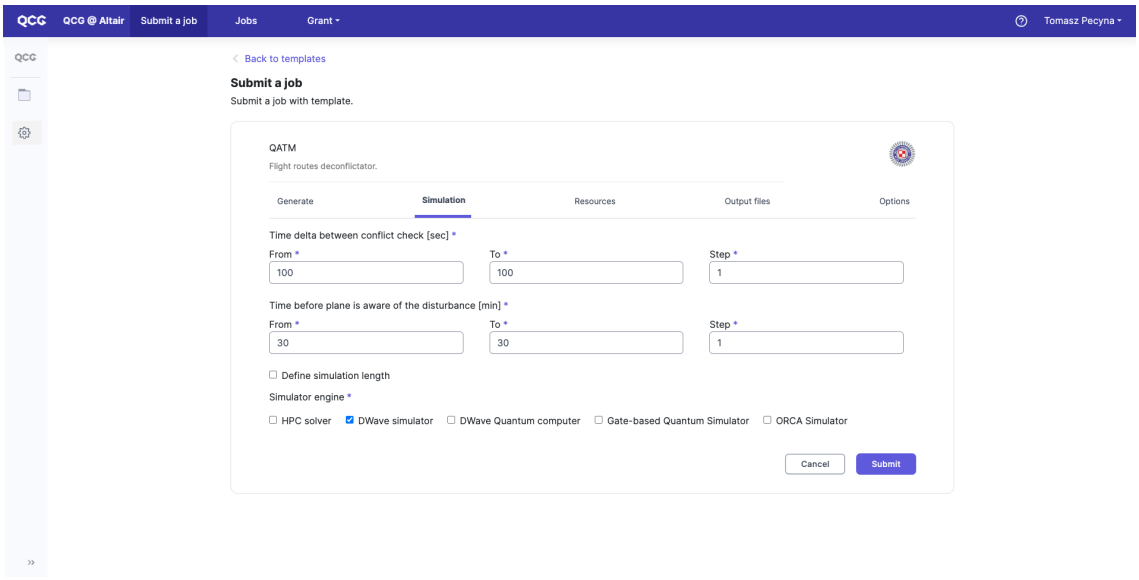


FIGURE 4.4: View of the QCG-Portal template for submitting the tactical aircraft deconfliction problem for solving on a quantum computer.

the research published in papers **P3** and **P4**, which are described in Sections 3.2 and 3.5, and also utilizes the QCG-QL tool (paper **P5**) described in Section 4.3.

The quantum backend is further divided into two submodules: the generator submodule and the deconfliction submodule. These submodules are integrated to operate in a loop. The generator creates air traffic scenarios with conflicts and generates possible routes for aircraft. These routes are then sent to the deconfliction submodule, which identifies conflicts and uses quantum hardware to select conflict-free routes that meet additional constraints. The process repeats, with the generator updating scenarios, adding emergencies (such as storms), and generating new routes until the simulation ends. The deconfliction submodule also allows for the assignment of weights to routes based on user preferences, enabling optimizations such as minimizing fuel consumption or prioritizing certain flights. This approach results in high flexibility in managing airspace situations and enables near-real-time analysis and conflict resolution.

The good practices established during the work on the EuroHPC PL project (see Section 4.1) facilitated the efficient design and implementation of the QCG-Portal view. The results of this work, specifically the submission view, are shown in Figure 4.4. This interface allows users to submit jobs on various quantum architectures, including the gate-based architecture, which is a part of this thesis. Additionally, users have the ability to generate instances of a chosen number of aircraft and maneuvers and to select control parameters such as feedback loop time.

The visualization system, SkyDodge, operates in replay mode and processes the entire air traffic scenario until its conclusion. The system can visualize aircraft routes, highlight conflicts, display selected alternative routes, adjust playback speed, rewind the scenario from a chosen time, filter subsets of aircraft, and present additional flight and sky condition information. It is designed to clearly show the actions of the controller or deconfliction system and enable detailed analysis of the decisions made. A screenshot of one potential conflict scenario and its resolution is shown in Figure 4.5.

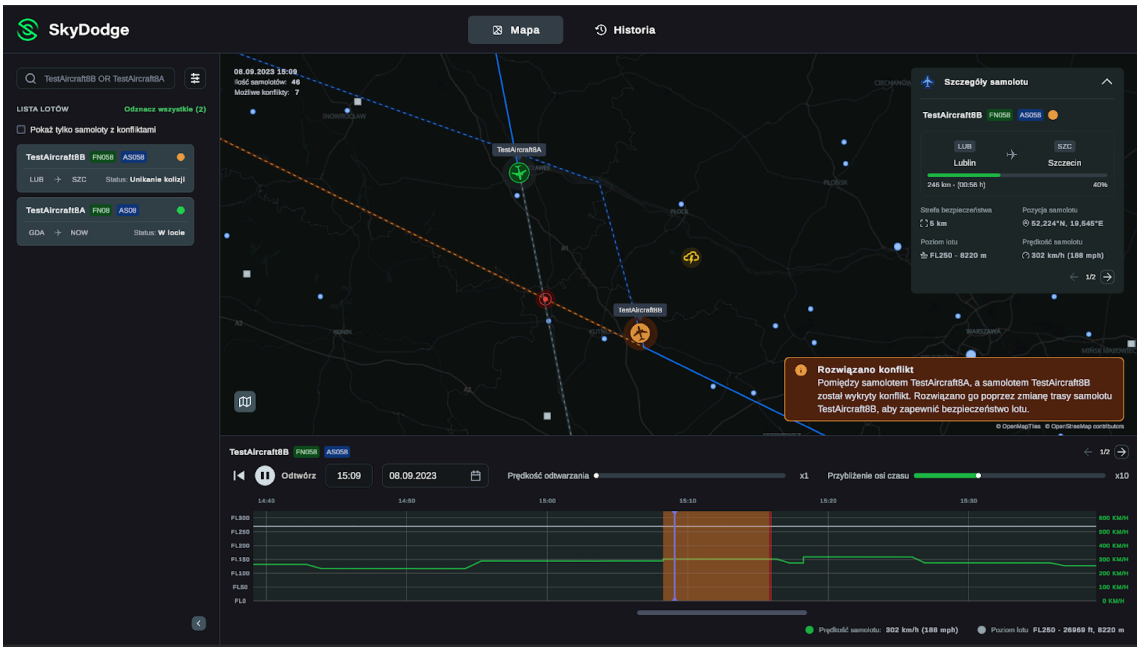


FIGURE 4.5: A screenshot of Skydodge module showing an exemplary conflict scenario and its resolution made by quantum hardware in the QATM project.

4.3 QCG-QuantumLauncher: a modular toolkit for managing quantum application scenarios

Initial experiments aimed at exploring the capabilities of quantum technology revealed a significant lack of high-level software solutions to facilitate conducting such experiments. Firstly, aside from the basic rules for Hamiltonian formulation in optimization problems, there was no tool available to automate this process, necessitating manual translation. This led to the development of an internal tool for translating logical Boolean functions into Hamiltonians, called Hampy. Secondly, for each different experimental approach, the entire workflow, including data reading, quantum backend selection, algorithm selection, results saving, etc., needed to be reimplemented. To simplify this process, a programming library called QCG-QuantumLauncher (QCG-QL) was developed, which eventually absorbed Hampy and evolved into a mature and modular solution for solving classical problems using quantum algorithms on quantum hardware.

The primary benefit of using QCG-QL, evident from the outset, is the significant simplification it provides in executing quantum algorithms to solve specific problems across various quantum machines. One notable advantage is the substantial reduction in the amount of code required. For typical experiments, QCG-QL reduces the code from several hundred lines to just a few. For instance, a typical implementation of QAOA solving JSSP on a gate-based quantum computer, including simple result saving, usually requires around 400 lines of code. Using QCG-QL requires writing only 5 lines of code.

The reduction would be less impressive if QCG-QL merely wrapped specific blocks of code into callable functions or classes. However, QCG-QL offers full modularity and flexibility. Its key advantage over similar quantum software libraries is the ability it gives quantum engineers to effortlessly switch between different quantum architectures. With just a single line of code, users can switch from a gate-based quantum computer to a quantum annealer or bosonic sampler. The same applies to switching between optimization problems or algorithms. All the required processes are handled seamlessly by QCG-QL, with minimal user input.

Although the tool was first publicly presented recently during the Parallel Processing & Applied Mathematics conference (**P5**), it had already been employed for internal use. QCG-QL has not only proven invaluable in conducting experiments for the papers **P1**, **P3**, and **P4** but it also serves as the quantum block for complete products and project outcomes described in Sections 4.1 and 4.2. With its high flexibility and extensibility, it enabled the implementation of a proof-of-concept integration of HPC and QC resources to develop a hybrid HPC-QC algorithm, which is described in detail in Section 4.4. Furthermore, QCG-QL serves as a foundation for future HPC-QC integration projects, such as EuroQCS-Poland[4]. Having demonstrated its value in multiple areas, QCG-QL can be regarded as an efficient tool for applying quantum computing approaches to solve optimization problems.

The implementation details of QCG-QL, along with the specific usage and description of its three core components, Backend, Algorithm, and Problem, are discussed in detail in paper **P5**.

4.4 QCG-QL extensions for solving combinatorial problems on hybrid HPC-QC systems

One particularly important area of interest recognized by the community is the development of hybrid HPC-QC algorithms. Quantum computers are unlikely to perform computations independently. Instead, classical and quantum computers will work in synergy. Quantum computers will accelerate certain HPC tasks, while quantum computations will, in turn, require HPC support for implementing specific algorithms [Callison and Chancellor, 2022].

For this specific, yet broad area, QCG-QL also demonstrates its applicability. In conjunction with QCG-PilotJob (QCG-PJ), a QCG service designed for the efficient execution of multiple tasks within a single allocation, it was employed to develop a hybrid algorithm that leverages HPC resources to accelerate the process of finding optimal variational parameters for the QAOA algorithm. Finding these optimal QAOA variational parameters is challenging and computationally expensive, as outlined in Section 3.3. In the approach mentioned, the computational load is uneven, with the majority of computations concentrated at the initial step, where parallel multipoint start search is used to optimize parameters for QAOA at a depth of $p = 3$. Moreover, the approach has a significant flaw in that it requires setting the number of initial points at the start of the algorithm. This can lead to inefficiencies, as unnecessary calculations may be performed if the optimal set of variational parameters is found early. Conversely, if the number of initial points is insufficient, the algorithm might fail to find the optimal variational parameters.

The new approach employing QCG-PJ together with QCG-QL operates as follows: To find optimal parameters for shallow circuits (with depth $p = 3$), multiple small processes are executed in parallel. If any process identifies parameters that yield lower energy than the current lowest, those parameters are interpolated, and all available resources are then focused on optimizing a longer circuit. At this point, no further small processes are initiated. Interpolation and optimization continue until the desired circuit depth is reached. However, if the interpolated parameters prove inadequate for deeper circuits, QCG-PilotJob triggers another set of small processes to search for new candidate parameters. The schematic view of this processing flow is depicted in Figure 4.6

This approach ensures efficient use of resources within the allocation, as deeper circuits typically require more computational power than shallower ones. Note, that each block optimizing variational parameters, as shown in Figure 4.6 represents a hybrid classical-quantum QAOA optimization. Since quantum computations are expected to become much faster than classical ones, access to

[4] `https://qt.eu/news/2022/european-quantum-computing-simulation`

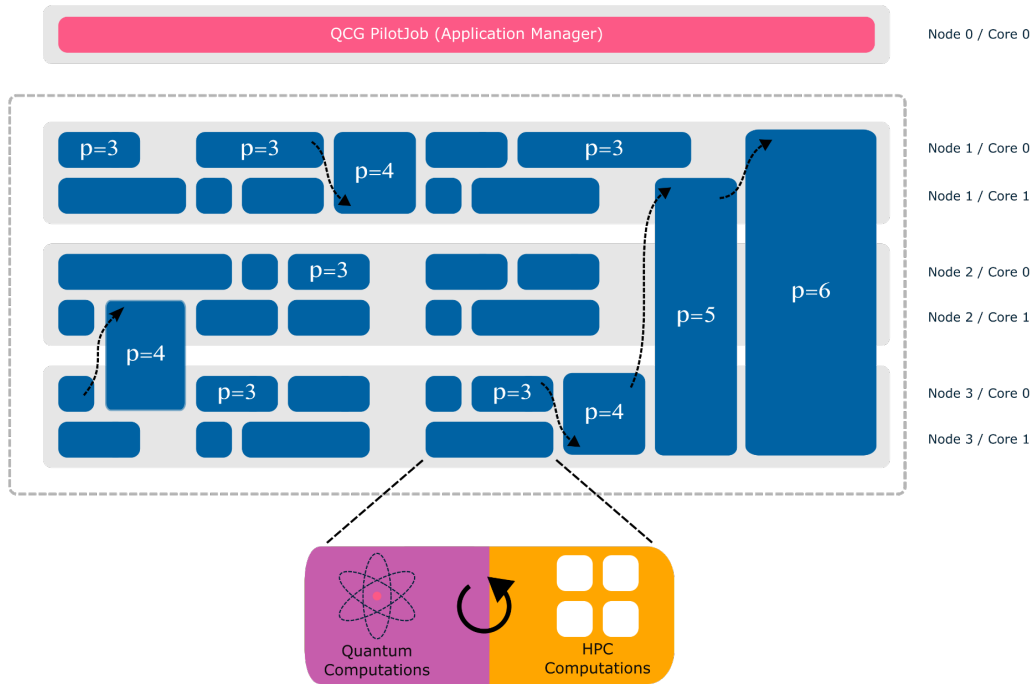


FIGURE 4.6: A schematic view of a processing flow of the hybrid classical-quantum model using QCG-PJ and QCG-QL. Each blue block represents a separate optimization process of a hybrid QAOA algorithm.

only a few quantum resources is sufficient to spawn dozens of processes. A quantum computer can be shared among classical processes and isn't idle during classical optimization, which would occur if each quantum resource were tied to a single classical processing unit.

Although large-scale hybrid HPC-QC computations are not feasible in the current NISQ era, the proposed approach lays the groundwork for such computations using quantum simulators, as demonstrated in the proof-of-concept solution. This approach was presented at the Messe München Laser World of Photonics workshop in 2023.

Chapter 5

Summary

This dissertation summarizes the work conducted during a 4-year PhD study. It begins by reviewing the current state of the field and diagnosing key issues. Subsequent research led to the development of specialized tools, which were then used to accelerate further investigations, creating a loop that culminated in real-world applications. Although not all challenges in the field are resolved, the thesis provides multiple advancements, particularly in terms of scientific knowledge, tools supporting users and applications, and the applications themselves, offering substantial and coherent contributions to the field. The scheme summarizing these advancements is depicted in Figure 5.1

The field of quantum computing suffers from insufficient interest in real-world applications of quantum approaches to optimization problems. Most existing solutions address problems with limited applicability, such as Max-Cut. This issue partially stems from the relative newness of quantum computing, but also from the high entry barrier for industry professionals, as there are no tools with a high level of abstraction to support performing experiments. The situation is further complicated by the existence of multiple quantum paradigms, each resulting in different quantum architectures, all of which behave differently and necessitate dedicated approaches.

The first knowledge gap addressed during the work on this thesis was experimenting with the widely used NISQ algorithm, QAOA. Specific Hamiltonians were formulated for the JSSP (**P2**), the EMVCP (**P1**), and the Tactical Aircraft Deconfliction Problem (**P3**). Experiments were conducted to investigate QAOA, including an analysis of educated guess strategies (**P2**), a comparison with quantum annealing (**P1**), and an examination of the relationship between energy and makespan in the measured JSSP schedules (**P2**). These experiments have demonstrated the practical application of quantum computing for solving real-world optimization problems. Additionally, they (**P1**, **P3**) marked the early internal use of QCG-QL and contributed to the design and implementation of the quantum backend for the EuroHPC PL portal interface, as well as the successful completion of the QATM project.

During the work, enhanced variants of QAOA were also examined. The purely quantum optimization enabled by the FALQON algorithm was found to outperform QAOA (Section 2.2.3), particularly for long circuits, which is not feasible with NISQ devices. The QAOAnsatz emerged as another promising alternative, demonstrating improved performance over QAOA by incorporating a subset of hard constraints into the Mixer Hamiltonian (**P3**). Conversely, the novel approach that contrasts with the QAOAnsatz by focusing on exploring more solutions also proved superior to the standard QAOA approach (**P4**). Moreover, this approach proved to be more powerful than the standard QUBO method commonly used for computations on quantum annealers. All of these accomplishments were, once again, greatly facilitated by the previously developed tools such as QCG-Hampy and QCG-QL.

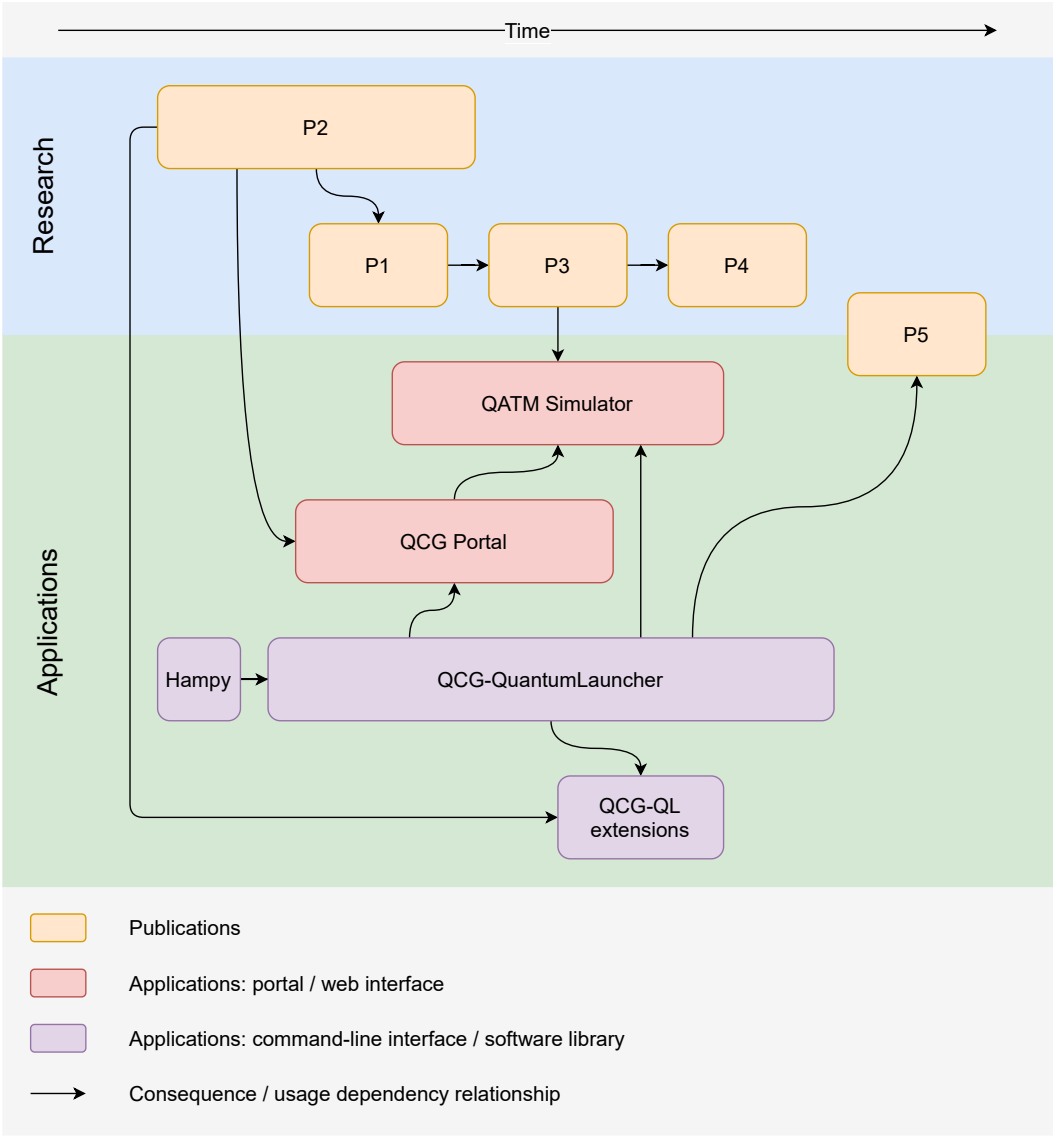


FIGURE 5.1: Summarized schema of the accomplishments described in this thesis. For clarity, the connections from QCG-QuantumLauncher to the papers **P1**, **P3**, and **P4**, which suggest its usage in experiments, are omitted.

As mentioned at the beginning of this section, the research work was interwoven with application development, creating a self-propelling loop. The main driving force of this loop was QCG-QL, which originated from the need for more efficient research (Section 4.3). The QCG-QL was then used to facilitate further research (**P1**, **P3**, **P4**), and ultimately became a subject of interest to a broader audience (**P5**). Firstly, QCG-QL solved the problem of code redundancy when performing many experiments with different problems, algorithms, or quantum computers. Secondly, QCG-QL provided a high-level abstraction layer between potential users and the implementation details, while still offering backdoors for more advanced researchers who might want to conduct more sophisticated investigations by allowing them to inherit from base QCG-QL classes.

QCG-QL was one of key software components in two projects, namely the EuroHPC PL project and the QATM project. The EuroHPC PL project (Section 4.1) concluded with the design and implementation of a platform for combinatorial optimization problems, where users interested in solving these problems could use a graphical interface for task definition, and easily access and

utilize available quantum devices provided and supported by PSNC. The platform was integrated with the QCG-Portal, enabling job organization, result analysis, and monitoring tools on PSNC computing resources. The successful completion of the EuroHPC PL project was primarily made possible by the foundational research on quantum combinatorial optimization, particularly the work presented in paper **P2**.

The QATM was the second project that concluded successfully, with the quantum backend being at the heart of the delivered solution (Section 4.2). The quantum solution described in **P3** allowed for tactical aircraft deconfliction using quantum algorithms, with possible optimization criteria such as minimizing fuel consumption or minimizing the number of trajectory changes for high-priority flights. Moreover, environmental factors such as storms and administrative border constraints were also taken into account. The quantum solution was implemented to run from the QCG-Portal being a natural enhancement of the solutions developed at PSNC. Moreover, a dedicated decision support tool called SkyDodge was developed to visually analyze the deconfliction process in replay mode, which could ultimately serve as a supporting tool for controllers performing manual tactical aircraft deconfliction. Enhanced experimental analysis done during the project led to another publication, **P4**

There are many avenues of research and application that this work could be extended to. However, two branches are of particular interest and are currently being explored. The first involves further investigation into relaxing problem formulations to simplify the Hamiltonian, thereby reducing the number of gates, which results in less noise impact and better measurement outcomes. The work presented in paper **P4** only touches on what might be possible, merely by substituting one-hot encoding with the NOT XOR function for the tactical aircraft deconfliction problem. The research plans include constructing a set of such possible substitutions and publishing a paper with precise guidelines on how to use these simplifications and relaxations, together with specific examples of optimization problems.

The second avenue for future work, this time concerning applications, is to design and implement an enhanced version of QCG-QL, tentatively titled QCG-QL2.0. This tool fits naturally into the future roadmap of PSNC and corresponds to upcoming projects that PSNC is set to participate in. QCG-QL2.0 would not only manage the use case of solving problems using algorithms on a quantum computer but would also handle more complex workflows, particularly multi-GPU and multi-QPU workflows. For this purpose, additional integration with QCG-PilotJob, a Python service that facilitates the execution of many tasks within a single SLURM allocation, would be necessary. Combined with user accounting, asynchronous execution, and hardware-agnostic interfaces, this would result in user-friendly, pythonic software, a general-purpose, scalable tool for researchers and engineers to launch any scenario involving quantum hardware with an arbitrary multi-platform hardware setup.

Additional paths for future research include, in no particular order, comparing the FALQON algorithm with the QAOA algorithm, investigating potential algorithms that facilitate synergy between HPC and quantum computing, applying for a project to advance the QATM project to a higher TRL, or reapproaching the tactical aircraft deconfliction problem to provide the model with fully quantum trajectory recovery or continuous-variable optimization.

Additional paths for future application include, in no particular order, integrating deconfliction capabilities within SkyDodge, integrating QCG-QL with QCG-PilotJob, and incorporating low-level hardware access and control into QCG-QL2.0.

Bibliography

[Abbas et al., 2023] Abbas, A., Ambainis, A., Augustino, B., Bärtschi, A., Buhrman, H., Coffrin, C., Cortiana, G., Dunjko, V., Egger, D. J., Elmegreen, B. G., et al. (2023). Quantum optimization: Potential, challenges, and the path forward. *arXiv preprint arXiv:2312.02279*.

[Ambainis et al., 2019] Ambainis, A., Balodis, K., Iraids, J., Kokainis, M., Prūsis, K., and Vihrovs, J. (2019). Quantum speedups for exponential-time dynamic programming algorithms. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 1783–1793. SIAM.

[Benioff, 1980] Benioff, P. (1980). The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of statistical physics*, 22:563–591.

[Bennett et al., 1997] Bennett, C. H., Bernstein, E., Brassard, G., and Vazirani, U. (1997). Strengths and weaknesses of quantum computing. *SIAM journal on Computing*, 26(5):1510–1523.

[Bernstein and Vazirani, 1993] Bernstein, E. and Vazirani, U. (1993). Quantum complexity theory. In *Proceedings of the Twenty-Fifth Annual ACM Symposium on Theory of Computing*, STOC '93, page 11–20, New York, NY, USA. Association for Computing Machinery.

[Bertsekas, 2014] Bertsekas, D. P. (2014). *Constrained optimization and Lagrange multiplier methods*. Academic press.

[Born and Fock, 1928] Born, M. and Fock, V. (1928). Beweis des adiabatensatzes. *Zeitschrift für Physik*, 51(3-4):165–180.

[Callison and Chancellor, 2022] Callison, A. and Chancellor, N. (2022). Hybrid quantum-classical algorithms in the noisy intermediate-scale quantum era and beyond. *Physical Review A*, 106(1):010101.

[Chuang et al., 1998] Chuang, I. L., Gershenfeld, N., and Kubinec, M. (1998). Experimental implementation of fast quantum searching. *Physical review letters*, 80(15):3408.

[Deutsch and Jozsa, 1992] Deutsch, D. and Jozsa, R. (1992). Rapid solution of problems by quantum computation. *Proceedings of the Royal Society of London. Series A: Mathematical and Physical Sciences*, 439(1907):553–558.

[Farhi et al., 2014] Farhi, E., Goldstone, J., and Gutmann, S. (2014). A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*.

[Feynman et al., 1957] Feynman, R. P., Vernon Jr, F. L., and Hellwarth, R. W. (1957). Geometrical representation of the schrödinger equation for solving maser problems. *Journal of applied physics*, 28(1):49–52.

[Grover, 1996] Grover, L. K. (1996). A fast quantum mechanical algorithm for database search.

[Hadfield, 2021] Hadfield, S. (2021). On the representation of boolean and real functions as hamiltonians for quantum computing. *ACM Transactions on Quantum Computing*, 2(4):1–21.

[Hadfield et al., 2019] Hadfield, S., Wang, Z., O'gorman, B., Rieffel, E. G., Venturelli, D., and Biswas, R. (2019). From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. *Algorithms*, 12(2):34.

[He et al., 2023] He, Z., Shaydulin, R., Chakrabarti, S., Herman, D., Li, C., Sun, Y., and Pistoia, M. (2023). Alignment between initial state and mixer improves qaoa performance for constrained optimization. *npj Quantum Information*, 9(1):121.

[Héder, 2017] Héder, M. (2017). From nasa to eu: the evolution of the trl scale in public sector innovation. *The Innovation Journal*, 22(2):1–23.

[IBM, 2023] IBM (2023). `https://newsroom.ibm.com/2023-12-04-IBM-Debuts-Next-Generation-Quantum-Processor-IBM-Quantum-System-Two,-Extends-Roadmap-to-Advance-Era-of-Quantum-Utility`. Accessed: 2024-09-10.

[Impagliazzo and Wigderson, 1997] Impagliazzo, R. and Wigderson, A. (1997). P= bpp if e requires exponential circuits: Derandomizing the xor lemma. In *Proceedings of the twenty-ninth annual ACM symposium on Theory of computing*, pages 220–229.

[Javadi-Abhari et al., 2024] Javadi-Abhari, A., Treinish, M., Krsulich, K., Wood, C. J., Lishman, J., Gacon, J., Martiel, S., Nation, P. D., Bishop, L. S., Cross, A. W., Johnson, B. R., and Gambetta, J. M. (2024). Quantum computing with Qiskit.

[Kitaev, 1997] Kitaev, A. Y. (1997). Quantum computations: algorithms and error correction. *Russian Mathematical Surveys*, 52(6):1191.

[Lubinski et al., 2023] Lubinski, T., Johri, S., Varosy, P., Coleman, J., Zhao, L., Necaise, J., Baldwin, C. H., Mayer, K., and Proctor, T. (2023). Application-oriented performance benchmarks for quantum computing. *IEEE Transactions on Quantum Engineering*, 4:1–32.

[Lucas, 2014] Lucas, A. (2014). Ising formulations of many NP problems. *Frontiers in Physics*, 2:5.

[Magann et al., 2022] Magann, A. B., Rudinger, K. M., Grace, M. D., and Sarovar, M. (2022). Feedback-based quantum optimization. *Physical Review Letters*, 129(25):250502.

[Nielsen and Chuang, 2001] Nielsen, M. A. and Chuang, I. L. (2001). *Quantum computation and quantum information*, volume 2. Cambridge university press Cambridge.

[Pelegrín and d'Ambrosio, 2022] Pelegrín, M. and d'Ambrosio, C. (2022). Aircraft deconfliction via mathematical programming: Review and insights. *Transportation science*, 56(1):118–140.

[Peruzzo et al., 2014] Peruzzo, A., McClean, J., Shadbolt, P., Yung, M.-H., Zhou, X.-Q., Love, P. J., Aspuru-Guzik, A., and O'Brien, J. L. (2014). A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5(1).

[Piontek et al., 2016] Piontek, T., Bosak, B., Ciżnicki, M., Grabowski, P., Kopta, P., Kulczewski, M., Szejnfeld, D., and Kurowski, K. (2016). Development of science gateways using qcg—lessons learned from the deployment on large scale distributed and hpc infrastructures. *Journal of Grid Computing*, 14:559–573.

[Preskill, 2018] Preskill, J. (2018). Quantum computing in the NISQ era and beyond. *Quantum*, 2:79.

[Schulz et al., 2022] Schulz, M., Ruefenacht, M., Kranzlmüller, D., and Schulz, L. B. (2022). Accelerating hpc with quantum computing: It is a software challenge too. *Computing in Science & Engineering*, 24(4):60–64.

[Shor, 1994] Shor, P. (1994). Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th Annual Symposium on Foundations of Computer Science*, pages 124–134.

[Venturelli et al., 2015] Venturelli, D., Marchand, D. J., and Rojo, G. (2015). Quantum annealing implementation of job-shop scheduling. *arXiv preprint arXiv:1506.08479*.

[Vikstål et al., 2020] Vikstål, P., Grönkvist, M., Svensson, M., Andersson, M., Johansson, G., and Ferrini, G. (2020). Applying the quantum approximate optimization algorithm to the tail-assignment problem. *Physical Review Applied*, 14(3):034009.

[Yarkoni et al., 2022] Yarkoni, S., Raponi, E., Bäck, T., and Schmitt, S. (2022). Quantum annealing for industry applications: Introduction and review. *Reports on Progress in Physics*, 85(10):104001.

[Zhang et al., 2019] Zhang, J., Ding, G., Zou, Y., Qin, S., and Fu, J. (2019). Review of job shop scheduling research and its new perspectives under industry 4.0. *Journal of intelligent manufacturing*, 30:1809–1830.

[Zhou et al., 2020] Zhou, L., Wang, S.-T., Choi, S., Pichler, H., and Lukin, M. D. (2020). Quantum approximate optimization algorithm: Performance, mechanism, and implementation on near-term devices. *Physical Review X*, 10(2):021067.

Publication reprints

Publication [P1]

Różycki, R., Józefowska, J., Kurowski, K., Lemański, T., Pecyna, T., Subocz, M., Waligóra, G.: A quantum approach to the problem of charging electric cars on a motorway. Energies. 16, 442 (2022). https://doi.org/10.3390/en16010442.

Ministry points / journal: 140

Contribution of authors:

- Rafał Różycki:
  - Co-authorship of the idea underlying the paper.
  - Co-authorship of the text of publication (Sections: Introduction, Problem Formulation, Quantum Approach)
  - Review and editing of the paper
- Joanna Józefowska:
  - Co-authorship of the idea underlying the paper
- Krzysztof Kurowski:
  - Co-authorship of the idea of the gate-based quantum approach
  - Co-authorship of the idea of the quantum annealing approach
- Tomasz Lemański:
  - Co-authorship of the text of publication (Sections: Introduction)
  - Consulting on the state of the art and energy storage solutions
- Tomasz Pecyna:
  - Co-authorship of the idea of the gate-based quantum approach
  - Implementation of the quantum gate-based approach
  - Performing experiments, results analysis
  - Co-authorship of the text of publication (Sections: Problem Formulation, Quantum Approach, Computational Experiment, Results, Discussion, Conclusions)
- Marek Subocz:
  - Co-authorship of the idea of the quantum annealing approach
  - Implementation of the quantum annealing approach
  - Performing experiments, results analysis

  - Co-authorship of the text of publication (Sections: Quantum Approach, Computational Experiment, Results)

- Grzegorz Waligóra:
  - Co-authorship of the text of publication (Sections: Introduction, Problem Formulation, Quantum Approach)
  - Review and editing of the paper

energies

MDPI

Article

A Quantum Approach to the Problem of Charging Electric Cars on a Motorway

Rafał Różycki [1,*], Joanna Józefowska [1], Krzysztof Kurowski [2], Tomasz Lemański [1], Tomasz Pecyna [1], Marek Subocz [2] and Grzegorz Waligóra [1]

[1] Institute of Computing Science, Poznan University of Technology, 60-965 Poznan, Poland
[2] Poznan Supercomputing and Networking Center, Institute of Bioorganic Chemistry of the Polish Academy of Sciences, 61-139 Poznan, Poland
* Correspondence: rafal.rozycki@cs.put.poznan.pl

Abstract: In this paper, the problem of charging electric motor vehicles on a motorway is considered. Charging points are located alongside the motorway. It is assumed that there are a number of vehicles on a given section of a motorway. In the motorway, there are several nodes, and for each vehicle, the entering and the leaving nodes are known, as well as the time of entrance. For each vehicle, we know the total capacity of its battery, and the current amount of energy in the battery when entering the motorway. It is also assumed that for each vehicle, there is a finite set of speeds it can use when traveling the motorway. The speed is chosen when entering the motorway, and cannot be changed before reaching the charging station. For each speed, there is given a corresponding power usage; the higher the speed, the larger the power usage. Each vehicle can only use one charger, and when its battery is full, the amount of energy is sufficient for reaching the outgoing node. We look for a feasible solution to the problem, i.e., a solution in which no vehicle has to wait for a charger. The problem is formulated as a problem of scheduling independent, nonpreemptable jobs in parallel, unrelated machines under an additional doubly constrained resource, which is power. Quantum approaches to solve the defined problem are proposed. They use the quantum approximate optimization algorithm and the quantum annealing technique. A computational experiment is presented and discussed. Some conclusions and directions for future research are given.

Keywords: electric motor vehicle; battery charging; power; energy; scheduling; parallel unrelated machines; quantum computing; quantum approximate optimization algorithm; quantum annealing

check for updates

Citation: Różycki, R.; Józefowska, J.; Kurowski, K.; Lemański, T.; Pecyna, T.; Subocz, M.; Waligóra, G. A Quantum Approach to the Problem of Charging Electric Cars on a Motorway. *Energies* **2023**, *16*, 442. https://doi.org/10.3390/en16010442

Academic Editors: Enrique Romero-Cadaval and Islam Safak Bayram

Received: 5 November 2022
Revised: 1 December 2022
Accepted: 27 December 2022
Published: 30 December 2022

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The interest in electric motor vehicles (EMVs) in the world is growing rapidly. According to the report of the Polish Alternative Fuels Association [1], from May 2021 to May 2022, the number of electric passenger cars in Poland increased by 81% (currently 48,883 vehicles), the number of electric buses increased by 43% (currently 762), and the number of motorcycles and scooters by 37% (currently 14,464). In total, approximately 26,000 more electric vehicles appeared on Polish roads than in the previous year. The number of generally accessible charging stations also increased by 47% (currently 2232). An additional premise for the increased interest in EMVs is the ban on the sale of new vehicles with internal combustion engines from the year 2035. It is one of the key elements of the implementation of the so-called European Green Deal, which aims, among other things, to achieve net-zero greenhouse gas emissions. Of course, to achieve this, it is necessary not only to reduce the number of vehicles emitting carbon dioxide, but also to increase the amount of energy obtained from renewable energy sources.

This paper focuses not so much on the emissivity, but on the comfort of EMV users. Facilitating the use and operation of a vehicle for an individual user may naturally contribute to a faster change in the preferences of purchasing a particular type of vehicle, and thus a

faster technological transformation in the field of transport. The problem of scheduling EMV battery charging is not new and has been discussed many times in many publications.

In [2,3], the problem of scheduling the battery charging process is considered. In the adopted model, it is assumed that during charging, the power consumption decreases linearly with time from a certain known initial value. The available power is a doubly constrained resource, in an amount not enough to charge all the batteries simultaneously. In [4], the use of an evolutionary algorithm to find the scheduling of drone-charging tasks in a multi-station charging station is considered.

In their article [5], Hahnsang and Kang propose the principles of planning activities aimed at extending the operation time and life of the battery. A weighted-k round robin (kRR) scheduling framework is designed. It consists of an adaptive filter and the *k*RR scheduler, whose task is to plan the number of parallel connected cells to be discharged while taking advantage of the recovery effect and also plan the load between k cells. The obtained result is up to 56% longer battery life and 50% more resistance to failures.

Mitici et al. [6] discuss problems related to planning flights by electric planes between airports. The authors develop an optimization model, thanks to which they create a flight schedule for electric aircraft. The schedule determines when the batteries are to be charged, the optimal charging time, when they are to be replaced with spare ones, and the optimal number of charging and replacement stations. The developed model enables three times more round trips than the size of the fleet. The included bi-linear charging profiles are divided into fast (up to 0–90%) and slow (90–100%) charging.

Xiaoqi et al. [7] consider the problem of battery replacement stations and charging the batteries at the stations. Their goal is to minimize charging costs while meeting the current demand for immediate battery replacement. In their research, they use the generalized Benders decomposition algorithm. The same algorithm is used by You et al. [8], who study the problem of planning the replacement of electric bus batteries.

Schaden et al. [9] considered the problem of charging at a single charging station with respect to current time-dependent electricity cost and vehicle state of charge. They found that the problem can be efficiently solved with only 1.5% error approximating concave power functions with piecewise linear functions.

In [10], Yang et al. described a model for choosing the best charging station on a highway for a single EMV. They found that using global information causes shorter waiting times than using only local information.

Del Razo et al. [11] proposed a modified version of the A* algorithm to schedule generated EMVs driving on a German highway connecting Berlin and Munich. Their work also enable dynamic changes to the schedule so that traffic can be accounted.

This article differs from the ones above mentioned, first of all, by taking a modern approach to solving a computationally difficult scheduling problem. This approach guarantees short solution times and is attractive due to the low cost of energy used to perform the calculations. The considerations are narrowed down to the section of the motorway where traffic flows only one way. On the modeled motorway section, there are many clearly ordered motorway junctions and a limited, usually small, number of charging points. The increase in the number of EMVs in use and long charging times pose the risk of large queues at charging points at stations. In the proposed approach, the aim is to prevent the arising of queues, and if this is impossible, to minimize their length or waiting times for charging to start. The proposed model is a relatively faithful reflection of reality, although, like every model, it includes some simplifications. Nevertheless, the scheduling problem formulated on its basis remains a difficult problem from a computational point of view. This means that to solve it, it is rational to use approximate algorithms, the effectiveness of which most often depends on the time of calculations, and thus on the amount of energy used to perform them. Therefore, in order to solve the considered problem, we propose to use quantum computers that can reveal their advantages over classical computers when used to solve problems of a combinatorial nature. Examples of such successful implementations of algorithms on quantum computers are reported in many

studies. Ajagekar and You [12] generally discuss many applications of quantum computers and compare them computationally to classical computers. In their research, they focus on planning energy systems and notice that quantum computers achieve better solutions in a much shorter time than classical computers. The authors emphasize, however, that not every optimization problem can be solved by quantum computers today. Faugler [13] notices the enormous potential of using quantum computing in the energy sector, where the complexity of computations is high and the dynamics of the modeled processes require quick adaptation to the existing situation. An important advantage of quantum computing is the possibility of building ecological energy management systems on its basis. Quantum computers were also used to calculate the charging of plug-in hybrid electric vehicles using the quantum annealing algorithm [14]. The quantum annealing method itself is becoming more and more popular and used for scheduling problems, e.g., nursing roster [15,16].

In this work, the hypothesis that quantum computers can be used to solve difficult, complex and real scheduling problems is verified with a specific example. The above hypothesis is tested experimentally with the use of two quantum computer architectures. The research was carried out on quantum computers from D-Wave Systems and IBM.

The paper is organized as follows. The problem formulation is given in Section 2. Section 3 presents our quantum approach. The computational experiment is described in Section 4, and its results are presented in Section 5. Section 6 is devoted to a discussion on the obtained results. Finally, some conclusions and final remarks are given in Section 7.

2. Problem Formulation

2.1. Problem Description and Parameters

In this section, we will formulate the problem of searching for the conflict-free order of charging vehicles on a chosen motorway section as a deterministic problem of scheduling jobs on machines. Let us start with a formal description of the problem and its parameters.

Given is a set of EMVs that are to drive through the motorway section, and need charging within this section. Each charging operation is divided into two phases:

- Phase I—reaching the charging point (station).
- Phase II—the process of charging the EMV's battery.

We define the motorway section M as a sequence of r nodes $M = (A, B, \ldots, X)$, where A is the start node, and X is the end node of the section. For each node its distance from the start node A is known, and denoted as $D_k^N > 0$, $k = 2, \ldots, r$, where, obviously, $D_1^N = 0$. We use the superscript N to distinguish between node distance and charging station distance, where we use S in the superscript. There are s charging stations over the section M. For each station $j, j = 1, 2 \ldots, s$, the following parameters are defined:

- D_j^S—distance between charging station j and the start node A.
- b_j—number of chargers at station j.
- B_{jl}—l-th charger at station $j, l = 1, 2, \ldots, b_j$.
- P_{jl}—available power of charger B_{jl}.

Furthermore, there are n EMVs on the motorway section M. We assume that we know the node where the EMV enters the motorway, the node where it leaves, as well as the time of entrance. We also know the total capacity of its battery, and the current amount of energy in the battery when entering the motorway. Moreover, we will assume that for each EMV, there is a finite set of speeds it can use when traveling the motorway. The speed is chosen when entering the motorway, and cannot be changed until reaching the charging station. For each speed, there is given a corresponding power usage; the higher the speed, the larger the power usage. Consequently, we can speak about driving modes, where a mode represents a relation between the speed of EMV and its power usage. Summarizing, for i-th EMV, $i = 1, 2, \ldots, n$, the following parameters are known:

- N_i^{in}—entrance (ingoing) node.
- N_i^{out}—outgoing node.
- a_i—arrival time of EMV i, i.e., the time of entering the motorway through node N_i^{in}.

- C_i^{full}—total capacity of the battery of EMV i.
- C_i^{curr}—current amount of energy in the battery of EMV i.
- m_i—number of driving modes "speed/power usage".
- $\mathbf{v}_i$—vector of available speeds, $\mathbf{v}_i = [v_{i1}, v_{i2}, \ldots, v_{il_i}]$.
- $\mathbf{p}_i$—vector of corresponding power usages, $\mathbf{p}_i = [p_{i1}, p_{i2}, \ldots, p_{im_i}]$.

As we can see, $C_i^{def} = C_i^{full} - C_i^{curr}$ is the energy deficit of EMV i, the amount of energy needed for its battery to be fully charged.

An example of data describing an EMV is shown in Figure 1.

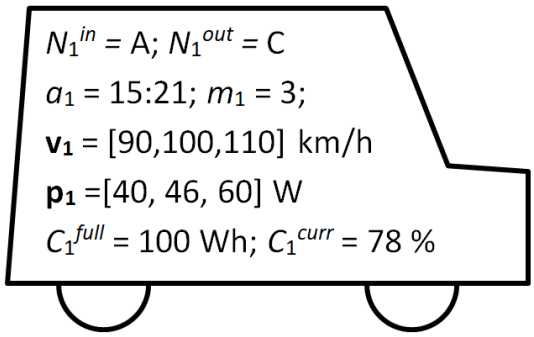


Figure 1. Parameters of an EMV.

Let us notice that the power usage function of an EMV, expressed by the "speed/power usage" modes, is nonlinear, and, usually, convex. If the function was linear, the energy consumed per distance unit as well as charging time would be both constant. In contrast, the more realistic, nonlinear power usage function does not assume constant distance/speed power usage, which, for example, might lead to the situation where, driving with a particular speed, the EMV may be unable to arrive at some stations because of the lack of energy. This indication, together with having all the data about the motorway infrastructure, can be utilized to define at the stage of processing a set of feasible (i.e., reachable) charging stations for each driving mode of an EMV. Notice that for each mode, this number may be different since the power usage function is nonlinear.

We further assume that the charging operations are independent, i.e., there are no precedence relations between EMVs, as well as nonpreemptable, i.e., the charging process in phase II cannot be interrupted (the battery is being charged without preemptions until it is full). We also make the following additional assumptions:

1. Each EMV can charge its battery only once, i.e., when the battery is full after phase II, the amount of energy is sufficient for reaching the outgoing node.
2. For each EMV, there exists at least one available speed for which the number of feasible charging stations is greater than 0.
3. Operations in phase I can be performed fully in parallel, i.e., we do not assume any limited capacity of the motorway, accidents, traffic jams, etc.
4. Each charging process is done by using exactly one charger.
5. The charging time in phase II is linearly dependent on the energy deficit C_i^{def} in the battery.

From the problem point of view, EMVs that can travel through the highway without charging are not of interest because they require no attention or alterations to the proposed solution. On the other hand, it would be tedious for EMV drivers to charge many times, even if it would result in reduced travel time. Therefore, we decided for a trade-off and, by assumption (1), allow only a single charging. Additionally, the assumption (2) assures that there are no instances for which we know already in the preprocessing step that there is no feasible solution. We further simplify the problem with assumptions (3) and (4). Current batteries charge linearly up to some threshold near being fully charged, after which the charging process slows down. As we noted in the assumption (5), we approximate it with

linear dependence; however, we are aware of the fact that the faithful reproduction of the charging process would eliminate some solutions.

We look for a feasible solution of the problem, i.e., a solution in which no EMV has to wait for a charger. To this end, for each EMV, we have to define its mode, i.e., speed and corresponding power usage (fixed between the entrance node and the charging point) and its charging point, i.e., charging station or charger which, we will discuss in points Sections 2.3.1 and 2.3.2 such that the time gap between the end of phase I and the beginning of phase II is equal to 0. To improve the readability of the rest of the paper, the term "EMV instance" will mean the assignment of both a specific driving mode and charging point to an EMV vehicle.

2.2. Classification of the Problem in the Classical Scheduling Theory

The problem formulated in Section 2.1 can be treated as a decision problem of scheduling (charging) jobs on parallel, unrelated machines. Each job corresponds to an operation of charging the battery of a single EMV. Jobs can be performed in various execution modes defined by pairs: available speed of the EMV and its corresponding power usage. Each machine corresponds to a single charging point. For each couple (machine, execution mode), the following two parameters can be calculated and set:

- Ready time r_i of job i, calculated as the sum of arrival time a_i of EMV i and the duration d_i^I of phase I for this EMV (i.e., the time needed for reaching the charging point): $r_i = a_i + d_i^I$.
- Execution time of job i, i.e., the duration d_i^{II} of phase II (the length of the charging process) for the corresponding EMV.

We look for such an allocation of jobs to machines (vehicles to charging points) that guarantees zero time gap between phase I and phase II for each EMV, i.e., no vehicle waits to start the charging process. The problem is NP-hard as a generalization of the problem of scheduling independent, nonpreemptable jobs in parallel, unrelated machines (see, for example, [17]).

A similar problem is considered in [18]. The authors study the unrelated parallel machine scheduling problem where the processing time of a job is based on resource allocation and the jobs are delivered in batches with unlimited batch capacity. A mathematical model is presented for minimizing the total weighted penalties of tardiness and earliness, resource allocation, and batch delivery costs with idle time and machine eligibility constraints. Three metaheuristic algorithms, including modified ant colony optimization (MACO), genetic algorithm (GA), and a hybrid algorithm based on the hybridization of MACO and GA, are proposed to solve the problem.

2.3. Charging Point Models

Let us now consider two options of defining the charging point. The applied model will then have an influence on, among others, the chance of finding a feasible solution.

2.3.1. Charger as a Charging Point

In the first, natural option, each charging point corresponds to a single charger with its unique parameters, in particular, the available charging power. Under this assumption, in an instance of the problem, the number of unrelated machines can be very large, which has a strong impact on the size of potential solutions and results in a huge computer memory requirements by the elaborated scheduling algorithm. An example of a motorway infrastructure in this case is presented in Figure 2.

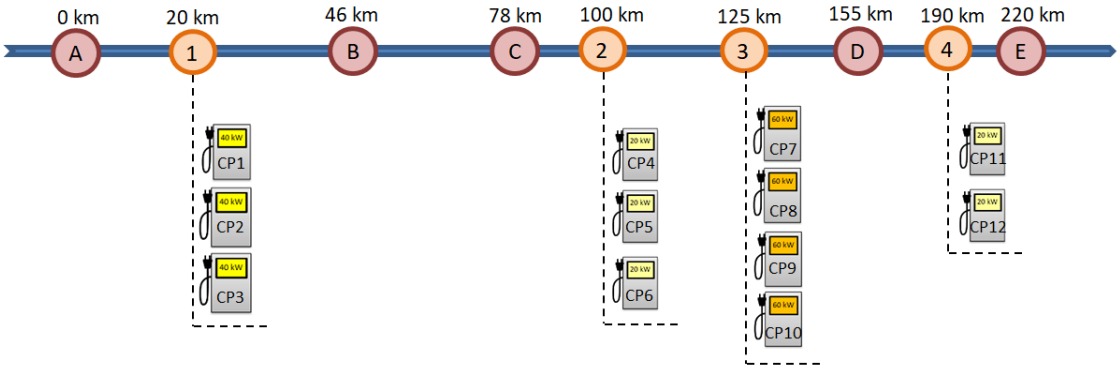


Figure 2. Example of a motorway infrastructure in model I.

2.3.2. Station as a Charging Point

A possible method of reducing the memory usage could be making some assumptions exceeding classical formulations of machine scheduling problems. Namely, if in the same location j (at the same charging station), there is a certain number of identical chargers ($P_{jl} = P_j$ for $l = 1, ..., b_j$), a set of such chargers may be treated as a single multi-functional machine able to perform several jobs in parallel. Such a machine is characterized by an additional parameter—the number of terminals (see parameter b_j in Section 2.1) defining the maximum number of jobs that can be processed simultaneously. Such an assumption can result in significant reduction of memory used by the scheduling algorithm; however, it may also reduce the chance of finding a feasible solution by the algorithm. It follows from that fact that the time of execution of a set of jobs by such a multi-functional machine is determined by the processing time of the longest jobs. As a result, some machines may not be fully loaded, and some idle times may occur.

An example of a motorway infrastructure in this case is presented in Figure 3.

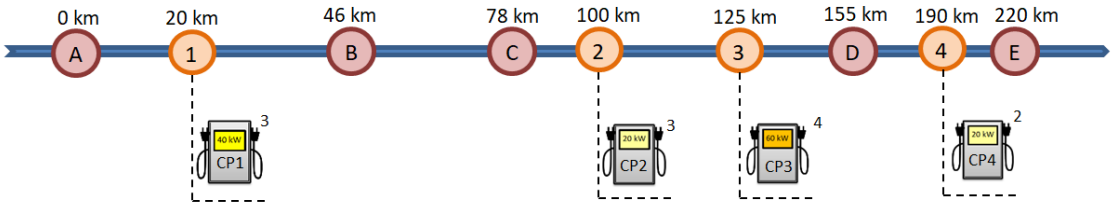


Figure 3. Example of a motorway infrastructure in model II with $b_1 = 3$, $b_2 = 3$, $b_3 = 4$, $b_4 = 2$.

3. Quantum Approach

Due to recent advancements in building real quantum hardware, especially in the past few years, quantum computation and quantum information have rapidly accelerated not only as a field of science, but also as a field of technology. Many researchers invest their time to make quantum computers useful in the current noisy intermediate scale quantum (NISQ) era, because they believe that quantum approaches can soon achieve significant advantages over classical computing in computation time, solution quality or energy usage. This is mainly due to exploiting superposition and entanglement phenomena. In short, having n qubits in superposition allows to perform simple computations simultaneously on up to 2^n possible states, while adding entanglement allows to make more complex computations, where two or more states depend on one another.

In the current NISQ era, there are two leading approaches which give exposure to the quantum realm: the gate-model architecture and the quantum annealing. The gate model paradigm is to use universal set of basis gates to encode any possible function [19]. Moreover, by the Solovay–Kitaev theorem, any arbitrary qubit gate can be approximated to some accuracy using the polylogarithmic number of these gates [20]. The nice properties do

not come with no cost however, as it remains a difficulty to build many-qubit fault-tolerant quantum computers. The most advanced in the technology seems to be the IBM company, which bases its quantum-chip technology on Josephson junctions cooled to near absolute zero temperatures. Currently reaching up to 127 qubits [21], they still suffer from noise, decoherence gate and measurement errors, which all lead to useless final measurements when the quantum circuit is too long.

The quantum-annealing-based devices are an attempt to build more powerful and accurate quantum computers at the cost of their universality. Their architecture restricts computation to a single type of algorithm—quantum annealing—which in and of itself is a technique with very broad applications. To make a problem solvable by quantum annealing, it has to be transformed to a QUBO formulation, which stands for quadratic unconstrained binary optimization. In short, it is a notation which assigns penalties to binary variables (representing qubits), as well as to pairs of them. Assigning rewards is also possible by setting a penalty to a negative value. We assume that the variable is selected when its respective qubit's state after measurement is 1. To implement problem constraints, we can assign penalties for selecting a single variable by controlling its superposition state to tilt toward value 0. A penalty for selecting two variables is implemented using entanglement. The bigger penalty there is for selecting a pair of variables; the more strongly their qubits are entangled, therefore the less likely they are to both be measured in the same state.

3.1. Conflict Avoidance Problem

An ideal situation on the motorway would allow every EMV to start charging its battery as soon as the vehicle arrives at the station. In that way, the users would waste no time waiting in the queue. Simultaneously, this would result in the chargers being used to their full potential, as less waiting time means more dense charging schedules.

In order to avoid waiting in queue, the problem we try to implement using a quantum computer is a conflict avoidance problem.

Due to the nature of the quantum computer, our approach will detect potential conflicts between each pair of vehicles, taking into account all their possible driving modes. By definition, conflicts only occur at specific charging points. At a single charging point, however, all potential conflict situations between each pair of vehicles using the motorway must be considered.

The number of potential conflicts between two vehicles i_1 and i_2 (at a given charging point), therefore, depends on the number of their driving modes m_{i_1} and m_{i_2}, respectively. Thus, at each charging point reachable by both EMVs, $m_{i_1} \times m_{i_2}$, different conflict situations should be checked. If a certain charging point was reachable by all EMVs traveling in any of the m driving modes, the number of possible different conflict matches (or lack thereof) to all vehicles would be $2^{n \times m}$. If you further consider that there are many such charging points, the exponent of this number is increased by an additional factor—number of charging points.

In order to find a feasible solution of the problem, for each EMV, it is necessary to choose such a driving mode and such a charging point that ensure that there are no charging conflicts along the entire stretch of motorway. If we denote by x_{ijk} a binary variable that takes the value 1 when the i-th EMV moving in k-th driving mode is charged at station j, a one-hot constraint should be fulfilled:

$$\sum_{j} \sum_{k} x_{i,j,k} = 1 \quad \text{for all } i = 1, 2, ..., n \tag{1}$$

Now we can move on to one of the key elements of our algorithm, which is the conflict matrix.

3.2. Conflict Matrix

The set of conflicts between each pair of vehicles moving in different driving modes is most conveniently represented in a binary array—the Conflict Matrix (CM). For a single

charging point, this array is a two-dimensional upper triangular matrix of the form shown in Figure 4.

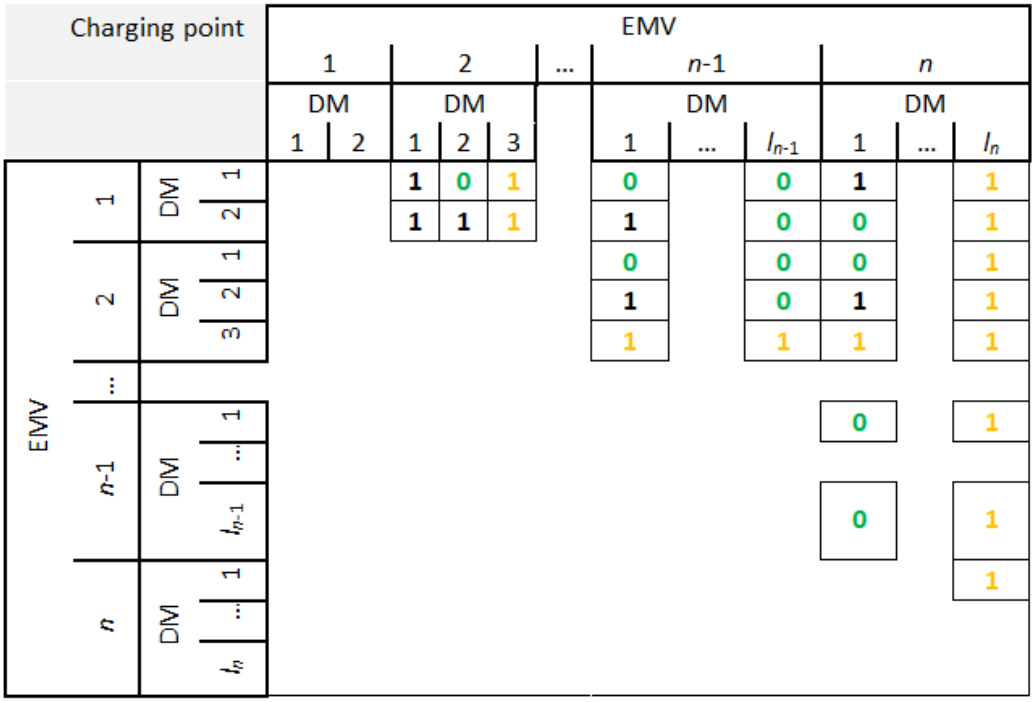


Figure 4. Excerpt of an exemplary 2D conflict matrix (single charging point).

The maximum size of the 2D CM can be calculated as

$$\sum_{i=1}^{n}\sum_{k=1}^{m_i} 1 \times \sum_{i=1}^{n}\sum_{k=1}^{m_i} 1 \tag{2}$$

Different types of conflict situations between two EMV's are depicted in Figure 4:

- Orange "1"—conflict because the charging point is unreachable by one of the EMVs moving in the selected driving mode;
- Black "1"—conflict because of charging at the charging point at the same time;
- Green "0"—no conflict.

Before the CM is fed to the quantum computer, all columns and rows representing driving modes leading to unreachable charging points can be removed. It allows to reduce the matrix size. Of course, the two-dimensional CM must be expanded to include a third dimension in which all charging points are represented. The 3D CM representing potential conflicts at all charging point is shown in Figure 5.

In the paper, we consider two different definitions of a charging point. The consequence of their application is discussed below.

3.2.1. Conflict Matrix for Charger as a Charging Point

The most general way to represent conflicts at each potential charging point is to include all individual chargers in the conflict matrix. We will call the so-constructed matrix the general conflict matrix (GCM). In this case, the solution to the problem is feasible when such a driving mode and charger can be found in the GCM for each EMV, guaranteeing the lack of conflict. The size of such a matrix is not greater than

$$\sum_{i=1}^{n}\sum_{k=1}^{m_i} 1 \times \sum_{i=1}^{n}\sum_{k=1}^{m_i} 1 \times \sum_{j=1}^{s}\sum_{l=1}^{b_j} 1 \tag{3}$$

This representation of conflicts has its advantages and disadvantages. Its advantage is that it allows detecting feasible solutions in which an EMV sequence is being charged on one of the chargers, even though charging is going on all the time on other chargers of the same station. Such a situation is shown in Figure 6.

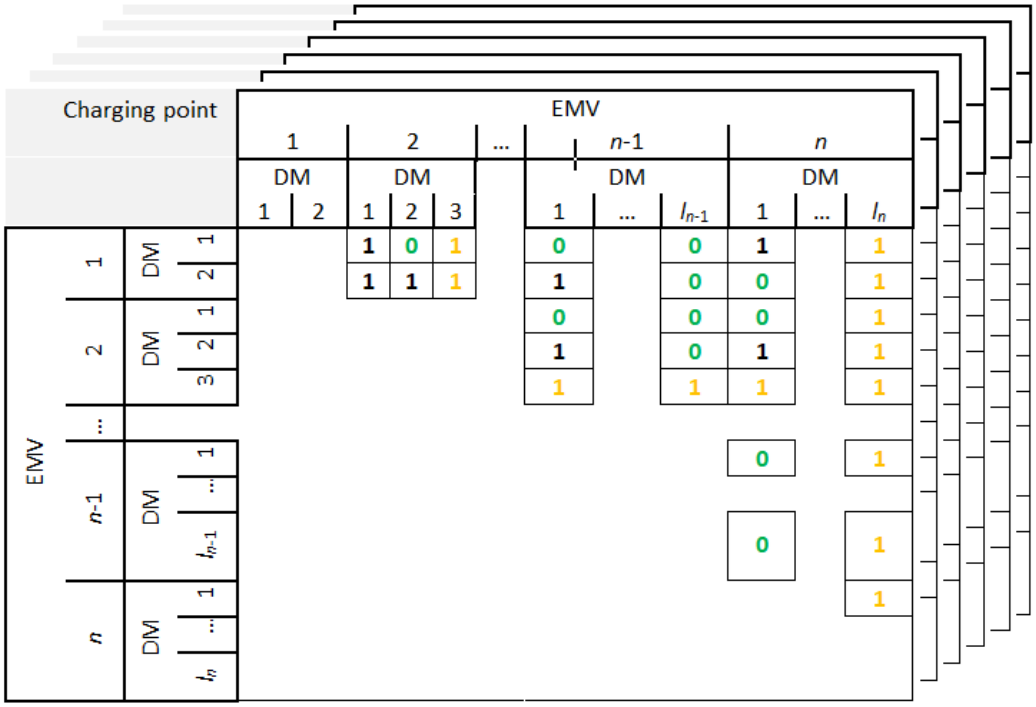


Figure 5. Excerpt of an exemplary 3D conflict matrix representing 5 charging points.

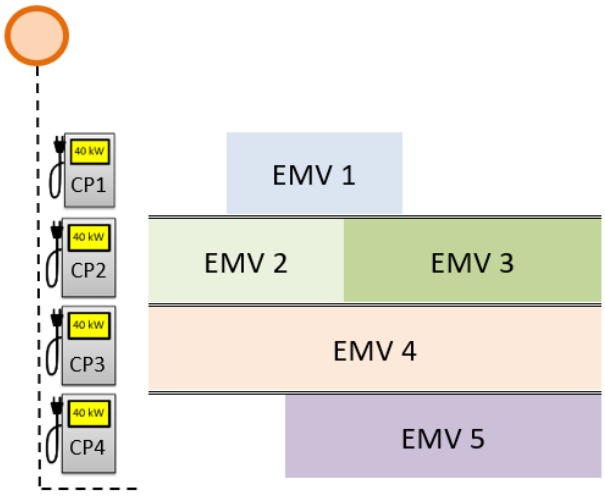


Figure 6. Gantt chart for "the charger as a charging point" case.

Its disadvantage, on the other hand, is that the 3D CM constructed in this way (GCM) unnecessarily contains redundant information about conflicts on chargers from the same location.

3.2.2. Conflict Matrix for Station as a Charging Point

Smaller 3D conflict matrices, and consequently fewer necessary calculations, can be obtained by taking into account the fact that some stations are equipped with multiple identical chargers. In this case, you can limit yourself to detecting conflicts at the station, aggregating information from individual chargers. In practice, this looks like counting the conflicts of pairs of EMVs at a given station. If each EMV at a station j containing b_j

chargers is in conflict with no more than $b_j - 1$ other EMVs, the solution is considered feasible. The size of the conflict matrix (in this case, called the station conflict matrix—SCM) so constructed is

$$\sum_{i=1}^{n}\sum_{k=1}^{m_i} 1 \times \sum_{i=1}^{n}\sum_{k=1}^{m_i} 1 \times s \tag{4}$$

This approach—while useful due to lower memory occupancy and shorter time of necessary calculations—does not allow to find some feasible solutions of the problem. The feasible solution in Figure 7 is treated as infeasible in this case, since EMVs 1, 4 and 5 are in conflict with more than three other EMVs.

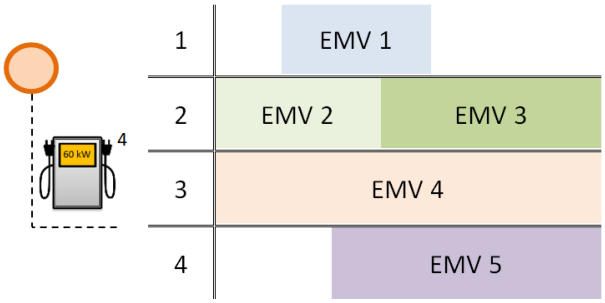


Figure 7. Gantt chart of a feasible solution which "the station as a charging point" rule treats as the infeasible one.

A feasible solution correctly recognized by "the station as charging point" rule is shown in Figure 8.

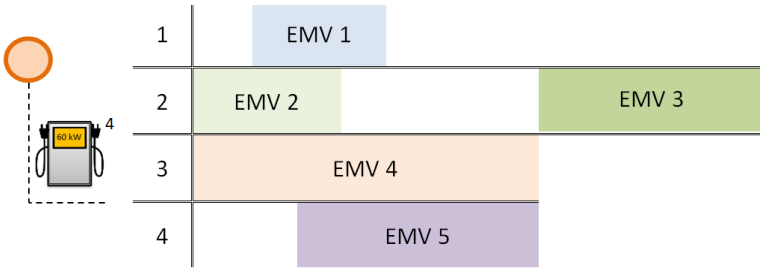


Figure 8. Gantt chart of a feasible solution for "the station as a charging point rule".

3.3. Gate-Based Approach and QAOA Algorithm

Mathematically a quantum state is typically represented as a superposition of the two basis states written in Dirac bracket notation:

$$|\psi\rangle = a|0\rangle + b|1\rangle \tag{5}$$

where the complex numbers a and b are called the amplitudes of the basis states satisfying normalization constraint $\sqrt{a^2 + b^2} = 1$. In contrast to classical computing, a qubit before measurement can be in any proportion between its basis states. After the measurement, a qubit collapses into one of them. The squared amplitudes describe the probabilities of a qubit to collapse into $|0\rangle$ or $|1\rangle$ after Z-basis measurement. This comes directly from the measurement and projection properties followed by their formal definitions [19]. A many-qubit state can be written using the tensor product. As a consequence, the number of base states and the number of amplitudes raise exponentially,

$$a_0|00\ldots00\rangle + a_1|00\ldots01\rangle + \cdots + a_{2^n}|11\ldots11\rangle \tag{6}$$

where n is the number of qubits, and we write $|00\ldots00\rangle$ as an abbreviation of a tensor product.

In quantum mechanics, the evolution of a quantum state can be always described with a unitary operator U

$$\psi' = U|\psi\rangle \tag{7}$$

This has some important consequences, e.g., that a quantum operation is always reversible. Strongly connected to the the unitary evolution is the notion of a Hamiltonian, which, in quantum mechanics, describes the total energy of a quantum system. In quantum computing, the Hamiltonian often acts as an expected value operator. Its size is $2^n \times 2^n$, where n is the number of qubits, and it can be interpreted as a map assigning a value to each of the 2^n basis states. The expected value of the Hamiltonian is called energy and can be written as

$$E = \langle\psi|H|\psi\rangle \tag{8}$$

From now on, we will use the term 'energy' to both indicate the expected value and the energy in batteries. Their meaning should naturally come from the context. Hamiltonians, together with parametrized unitaries, are often used in a special family of hybrid quantum–classical algorithms, which are able to solve combinatorial optimization problems, namely variational algorithms. The approach is to construct such a circuit and find such parameters so that the expectation value of a user-defined Hamiltonian is minimum. The user-defined Hamiltonian acts as a function which aggregates both the cost function as well as the constraints.

To tackle the problem of conflict-free EMV charging, as a hybrid quantum–classical variational algorithm, we use the quantum approximate optimization algorithm (QAOA) [22]. QAOA is an algorithm that tries to approximate continuous-time quantum adiabatic evolution [23], which states that if we change the time-dependent Hamiltonian slowly enough, a system will remain in its eigenstate. Mathematically, the adiabatic evolution can be written as

$$H(t) = (1-t)H_M + tH_C \tag{9}$$

In QAOA, we often call H_M a mixer Hamiltonian, and H_C is a cost Hamiltonian. H_M is usually composed of Pauli-X gates so that the quantum state can be easily prepared as an equal superposition of the basis states using Hadamard gates. The slow enough evolution is performed using Trotter–Suzuki approximation, and hence the final form of QAOA is

$$|\psi_p(\overrightarrow{\gamma}, \overrightarrow{\beta})\rangle = e^{-i\beta_p H_M} e^{-i\gamma_p H_C} \dots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_C} |+\rangle^{\otimes n} \tag{10}$$

The $|+\rangle$ denotes equal superposition, and p is a key parameter of QAOA, which defines the length of the circuit, and thus the approximation quality. The longer the circuit is, the better the solution can be, but it is harder to optimize the variational parameters β and γ. This trade-off is especially visible when working on a real NISQ hardware, as the noise prevents from taking advantage of longer circuits.

We can see that the role of H_C in QAOA is twofold. Firstly, by the adiabatic theorem, it defines the building blocks of the circuit. Secondly, by being the expected value operator, we can treat it as a function to minimize.

3.4. Gate-Based Hamiltonian Formulation

Due to technological limitations, we will limit the implementation of the Gate-based approach only to the "station as a charging point" model. We established the one-hot constraint (1) in Section 3.1. Additional constraints arise from the need to avoid conflicts and are based on information stored in the SCM. Since the cost Hamiltoniain H_C is a function that aggregates constraints, we need to convert them accordingly. Every Boolean function can be represented as a sum of Pauli-Z clauses [24]. Following the recipe, we can write the one-hot constraint as a *1-in-n* function, or rather its negation, as we are minimizing the expected value, selecting only one instance of the same EMV

$$H_{\text{one-hot}} = \sum_i 1\text{-}in\text{-}n(x_{i,1,1}, \dots, x_{i,s,l_i}) \tag{11}$$

where 1-*in*-*n* is derived with the Fourier transform as in [24].

Likewise, we define the conflict Hamiltonian. Firstly, let us define a set containing all EMVs possibly conflicted with a EMV i driving in mode j and charging at a station k

$$S_{i,j,k} = \{x_{i'j'k'} : \ SCM(i,j,k)(i'j'k') = 1\} \tag{12}$$

Since we are interested only in those situations where the number of EMVs simultaneously charging at a station is greater than available chargers, we define a collection S containing only these $S_{i,j,k}$ with cardinality greater than b_j

$$S = \{S_{i,j,k} : \ |S_{i,j,k}| > b_j\} \tag{13}$$

To prevent more than b_j EMVs to arrive at a station at the same time, it is sufficient to prevent $b_j + 1$ EMVs to arrive, so we define a set P of such possible combinations:

$$P = \left\{ \binom{s}{b_j} \forall s \in S \right\} \tag{14}$$

The set P can be easily interpreted as a sum of logic AND functions preventing more than b_j EMVs to simultaneously charge at that station:

$$H_{\text{conflict}} = \sum_{p \in P} \frac{1}{2^{|p|}} \prod_{x_{i'j'k'} \in p} (I - Z_{i'j'k'}) \tag{15}$$

where $Z_{i'j'k'}$ is applied on a qubit assigned to $x_{i'j'k'}$.

Having the constraints converted, we can construct the final Hamiltonian as

$$H_C = H_{\text{one-hot}} + H_{\text{conflict}} \tag{16}$$

3.5. Quantum Annealing

Quantum-annealing-based solutions also utilize Hamiltonians to formulate optimization problems. The problem's energy landscape is described by the following equation:

$$H = \sum_i \mathbf{Q}_{i,i} \mathbf{x}_i + \sum_{i<j} \mathbf{Q}_{i,j} \mathbf{x}_i \mathbf{x}_j \tag{17}$$

where $Q_{i,j}$ are entries of upper-triangular $\mathbf{Q}$ matrix (18) and x_i is a state i-th qubit ended up in after measurement. The $\mathbf{Q}$ matrix is used to control the degree of qubit superposition imbalances and entanglement strengths.

$$\begin{bmatrix} Q_{11} & Q_{12} & \cdots & Q_{1n} \\ 0 & Q_{22} & \cdots & Q_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & Q_{mn} \end{bmatrix} \tag{18}$$

The upper-triangular matrix $\mathbf{Q}$ is constructed from the state Hamiltonian.

Every entry on the main diagonal of the $\mathbf{Q}$ matrix describes how much energy the solution will gain if a given variable is selected. The selection of a variables is in fact a qubit being measured as 1. Every other entry of the $\mathbf{Q}$ matrix describes how much energy the solution will gain if two qubits end up in the same state. For example, if $Q_{i,j} = 2$, the solution's energy will rise by 2 if both x_i and x_j are selected. Positive values in $\mathbf{Q}$ matrix act as penalties for the model. It is also possible to assign negative penalties to encourage the QPU to select a given variable.

3.6. Quantum Annealing Hamiltonian Formulation

After the GCM (or SCM) is created, we apply several constraints to quantum variables, ensuring that the final result represents a feasible solution to the EMV conflict avoidance problem. Every qubit represents a selection of its corresponding EMV instance. The qubit representing j-th instance of i-th vehicle is denoted as $q_{i,j}$.

The first restriction applied to qubits is a one-hot constraint (1). We group all qubits describing instances of the same EMV, sum their values and require that their sum be equal to one. This way, the selection of just one EMV instance (charging point and driving mode) is ensured:

$$H_{\text{one}-\text{hot}} = \sum_{i=1}^{n} \left((1 - \sum_{j=1}^{s} \sum_{k=1}^{m_i} \sum_{l}^{b_j} x_{i,j,k,l})^2 \right) \tag{19}$$

where n is a number of vehicles, m_i is a number of driving modes of vehicle i, s is a number of available stations, and b_j is a number of chargers available on station k. Every time the QPU chooses more (or less) than one instance of every EMV, the solution's energy increases exponentially.

The second part of the problem Hamiltonian ensures the minimization of conflicts during charging. The equation differs depending on whether we treat each charger as a separate charging point:

$$H_{\text{conflict}} = \sum_{j=1}^{s} \sum_{l=1}^{b_j} \left(\sum_{i=1}^{n} \sum_{k=1}^{m_i} \sum_{i'=i+1}^{n} \sum_{k'=k+1}^{m_i} x_{i,(j,l),k} * x_{i',(j,l),k'} * \text{GCM}(i,(j,l),k)(i',(j,l),k') \right) \tag{20}$$

or each station as a collection of chargers:

$$H_{\text{conflict}} = \sum_{j=1}^{s} \left(\sum_{i=1}^{n} \sum_{k=1}^{m_i} \sum_{i'=i+1}^{n} \sum_{k'=k+1}^{m_i} x_{i,j,k} * x_{i',j,k'} * \text{SCM}(i,j,k)(i',j,k') \right) > b_j \tag{21}$$

Note that Equation (21), for the cost of being less precise, contains fewer variables, which means greater problems can be embedded on QPU chips:

$$H_C = H_{\text{one}-\text{hot}} + H_{\text{conflict}} \tag{22}$$

The final Hamiltonian incorporates both one-hot constraint and conflict constraint. It is possible to apply weights to those Hamiltonian in order to regulate their impact, but experiments showed that an equal representation results in a robust optimization.

3.7. Note on Energy

Looking at the Hamiltonians defined in both quantum techniques, we can see that they correlate to the number of constraints that failed to be met. Even though the energy value is not proportional to the number of unresolved conflicts, it can be interpreted as such. The key point is to understand that minimizing the Hamiltonian energy directly improves the solution.

4. Computational Experiment

4.1. Assumptions

To confirm the usefulness of quantum approaches for solving the practical problem considered in this work, a number of computational experiments were carried out. The assumptions for the experiments were matched to the current strong technological limitations (small number of available qubits) of available quantum computers. More attractive due to the size of the possible instances of the problem is the DWave quantum computer, so more experiments were carried out for the QUBO algorithm. A more general conflict matrix, GCM, and "charger as a charging point" model, were used in all experiments for this case. A test instance generator was implemented specifically for the purpose of these experiments.

The second quantum technology presented above—the gate-based architecture (and QAOA algorithm)—was tested only on an instance limited in size. Due to the small capacity of currently available gate-model-based quantum computers, only one set of toy-size experiments was conducted. The experiments instance included part of the A4 motorway (described below), but only 5 EMVs were considered. Moreover, we decided to utilize a smaller version of the conflict matrix, SCM, and consequently the "station as charging point" method for the conflict avoidance problem.

All experiments assumed a convex function of EMV power usage while driving—an assumption that reflects the actual dependence of power consumption on vehicle speed. In the experiment for the real section of the motorway, precise tabular data specified for two specific EMV models were adopted. As for the available EMVs types, in the experiments with QAOA, we follow ref. [25], wherein two types of vehicles are described: Mitsubishi i-MiEV with 16kWh battery capacity and Nissan Leaf with 24 kWh battery capacity.

However, due to the limitations of quantum computers, a relatively small number of driving modes were assumed in all experiments. For example, it resulted in only 18 EMV instances (for 5 EMVs) in the QAOA experiment.

4.2. Test Instances—Generator

As for a dataset for our implementation of the QUBO algorithm, we decided to write our own instance generator. The process of generating a complete instance comprises three steps. Firstly, the motorway is generated. The key parameters of the motorway are its length, number of stations (or alternatively, station frequency), number of nodes (alternatively, their frequency) and chargers in each station. Secondly, EMV types are generated. Each EMV has different battery capacity ($C_i^{full} \in [10, 30]$ kWh, $i = 1, 2, ..., n$) as well as an individual vector of power usages corresponding to global vector of available speeds ($p_{im_i} \in [80, 250]$ kWh/km, $i = 1, 2, ..., n$). We set the vector so that the energy consumption raises with the EMV speed. In the last step, EMVs entering and leaving the motorway are randomly selected. Their amount of energy when entering a motorway is also randomly selected. In this step, we can control the difficulty of an instance by changing two parameters: density, which defines the overall occupation of charging station and simulation time, which defines size of the instance. The range of parameter values that affect the size of GCM is given below:

- $s \in \{1, 2, ..., 25\}$
- $b_j \in \{2, 3\}$, $j = 1, 2, ..., s$
- $n \in \{3, 4, ..., 33\}$
- $m_i = 6$, $i = 1, 2, ..., n$.

The instance generator is designed so that every instance is guaranteed to have at least one feasible solution. If all the EMVs travel with their lowest speed, there is always a schedule such that there is no conflict at the charging stations. Note that this does not mean that there are other feasible solutions in which EMVs can travel with higher speed.

4.3. Exemplary Practical Instance

Our instance generator allows for the generation of numerous instances of different kinds. The problem of choosing an optimal location for the charging infrastructure is interesting in itself, and has been widely studied, e.g., [26]. Since we are considering only motorways, we can make an assumption that gas stations will be naturally transforming themselves into charging stations, as the basic infrastructure, such as exits or parking spaces, are already there.

In order to get closer to real data, in some experiments, we will be considering the A4 motorway, which is currently the longest motorway in Poland (669 km) and it is a part of the European route E40. We will be considering a section of the motorway between two big polish cities: 104 km route in direction from Katowice to Krakow. The section characteristics [27] are as follows:

- $r = 18$ nodes
- $s = 4$ gas stations which we will interpret as charging stations.
- We will assume each charging station has 2 terminals in total.

4.4. Runtime Environment

All of the experiments with the QUBO algorithm were obtained using D-wave's Advantage 4.1 computer with the pure binary quadratic model (BQM) representation. This means that the entirety of the problem's solution computation was performed on a quantum computer. The only preprocessing done was the problem's embedding on an Advantage chip and mapping problem variables to qubits.

The experiments for QAOA were run on a 27-qubit quantum machine: ibmq_toronto (32 Quantum Volume, 1.8 k circuit layers operations per second).

5. Results

Two sets of instances were used in our experiments with QAOA. To acquire Figures 9–11, we used artificially constructed (using the implemented generator) instances of motorways with varying number of EMVs utilizing the motorway. Figure 12 represents data from a part of the A4 motorway located in the south of Poland.

The height of the bars of figures is an average value from different instances of the problem, grouped by the number of cars (Figures 9 and 12) or the size of the general conflict matrix (Figure 10) As seen in Figures 9 and 10, the quality of the solution depends on the instance's number of EMVs and the size of the corresponding GCM. This result is hardly surprising, as number of qubits used in the computation is determined by the size of the GCM.

In Figure 11, the distribution of the dependence of the GCM size on the value of the parameter n (the number of EMVs) is presented. It confirms that the number of EMVs is highly correlated with the GCM size in our experiments.

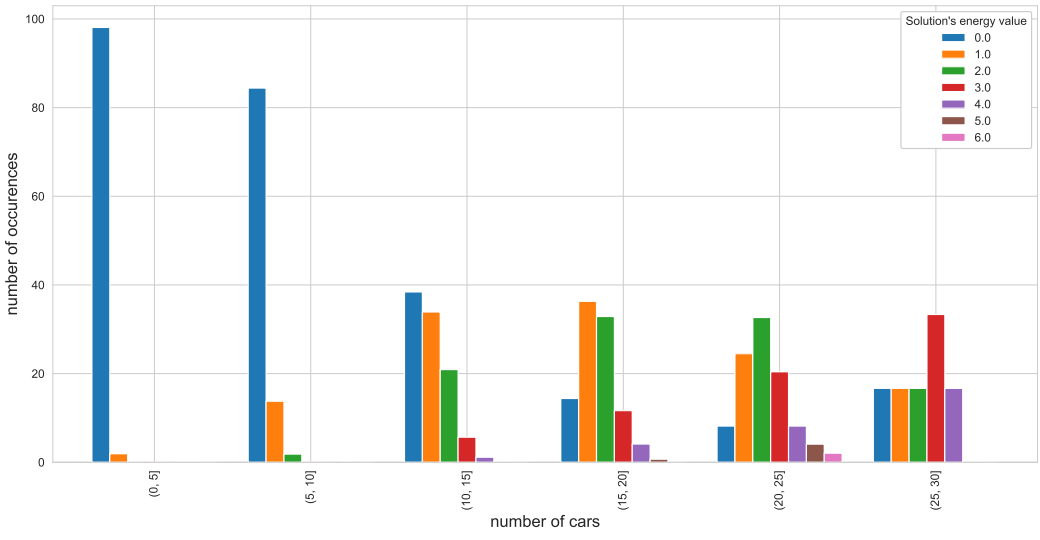


Figure 9. The quality of quantum-annealing-based solutions to the conflict-free EMV charging problem with respect to the number of EMVs in the problem instance. Different colors represent different number of unresolved collisions occurring in the solution.

In addition to data on the efficiency of the tested algorithm in finding solutions to the considered problem, the computation times on the quantum computer seem equally interesting. Below, we present the mean timing values:

- QPU_SAMPLING_TIME: 1.97 s
- QPU_ANNEAL_TIME_PER_SAMPLE: 20.0 µs
- QPU_READOUT_TIME_PER_SAMPLE: 156.20 µs

- QPU_ACCESS_TIME: 1.98 s
- QPU_ACCESS_OVERHEAD_TIME: 107.13 ms
- QPU_PROGRAMMING_TIME: 15.07 ms
- QPU_DELAY_TIME_PER_SAMPLE: 20.54 µs
- POST_PROCESSING_OVERHEAD_TIME: 1.12 ms
- TOTAL_POST_PROCESSING_TIME: 8.59 ms.

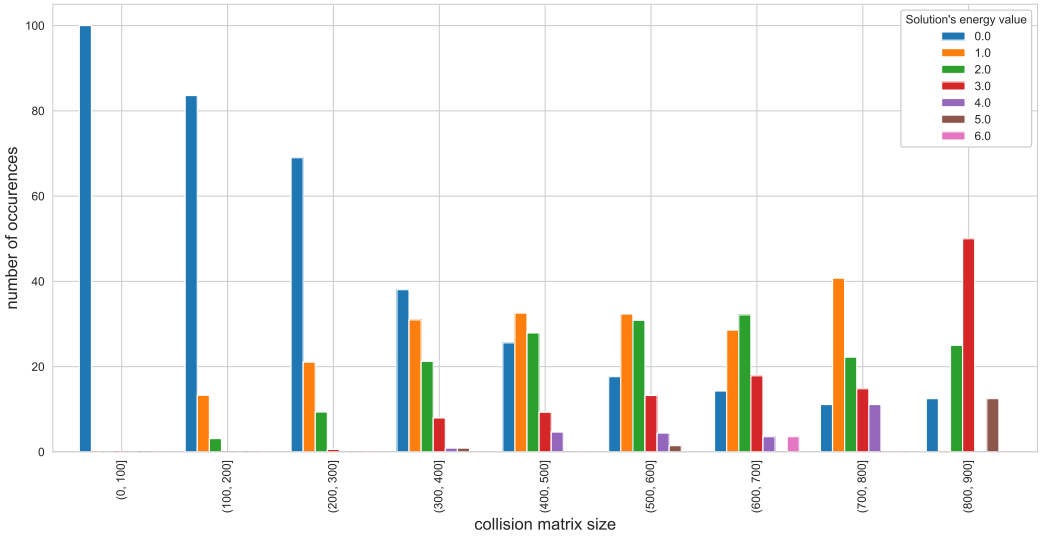


Figure 10. A figure equivalent to Figure 11 with respect to CM size instead of number of EMVs. The lower the solutions' energy value, the better.

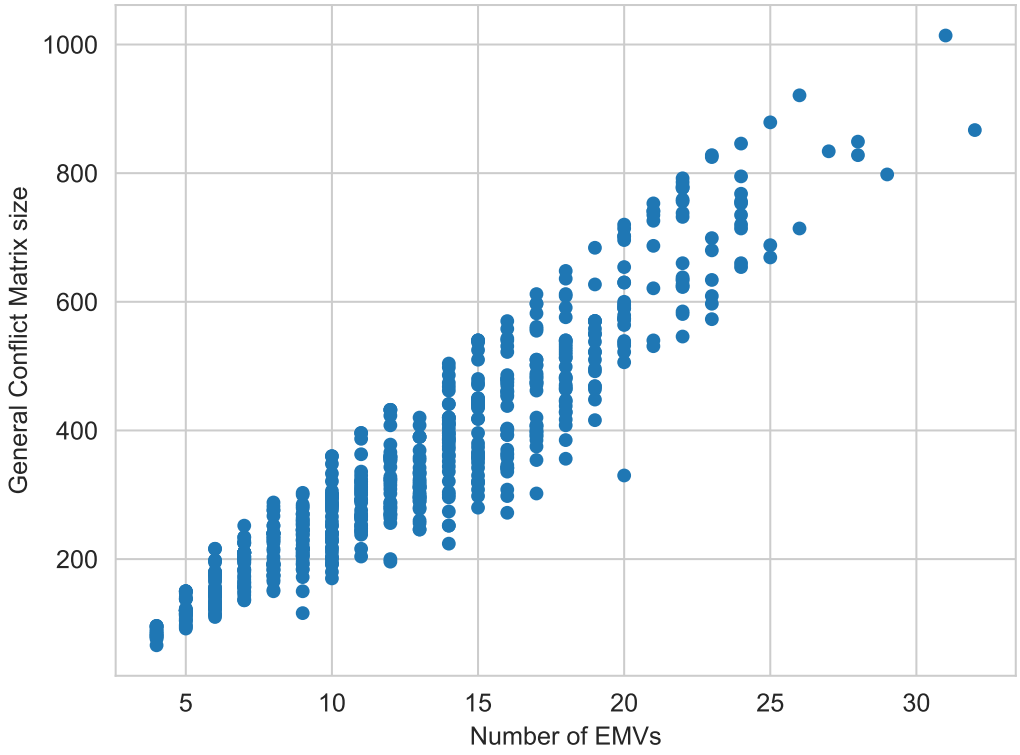


Figure 11. The size of the GCM size with respect to the number of EMVs in a problem instance.

The values were obtained directly from the D-wave's leap platform. The precise meaning of those metrics may be found in [28], where many helpful tutorials and guides are included.

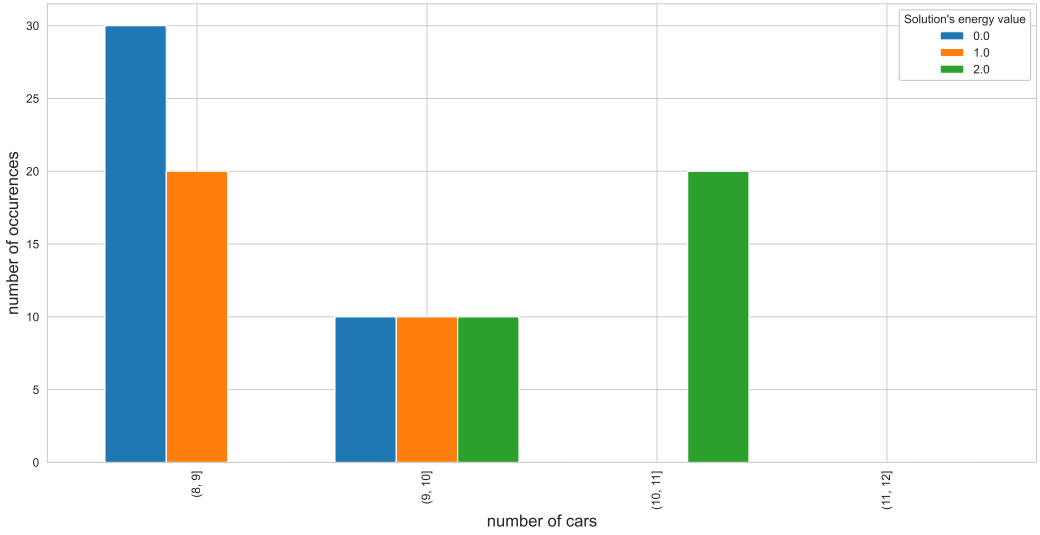


Figure 12. The quality of quantum-annealing based solutions to the conflict-free EMV charging problem with respect to the number of EMVs present on the A4 motorway. Different colors represent different numbers of unresolved collisions occurring in the solution.

In the experiments with a gate-based model of a quantum computation (QAOA), we tried to study the effect of the length of the circuit on the chance of finding feasible solutions to the considered problem. We show the results for QAOA for different circuit length (from $p = 1$ to $p = 5$) in Figure 13. We can see that the highest chance of measuring optimal solution is when using circuit of length 2. After that, the noise issues disturb the optimization process. We can also observe, that for $p = 2$, the probability of measuring the feasible solution is around 0.5%, which is several times better than drawing a random bitstring. The computational times (classical and quantum computation) for different circuit length are the following:

- $p = 1$: 1 h 7 m 7 s
- $p = 2$: 1 h 10 m 59 s
- $p = 3$: 1 h 19 m 3 s
- $p = 4$: 1 h 29 m 31 s
- $p = 5$: 1 h 35 m 29 s.

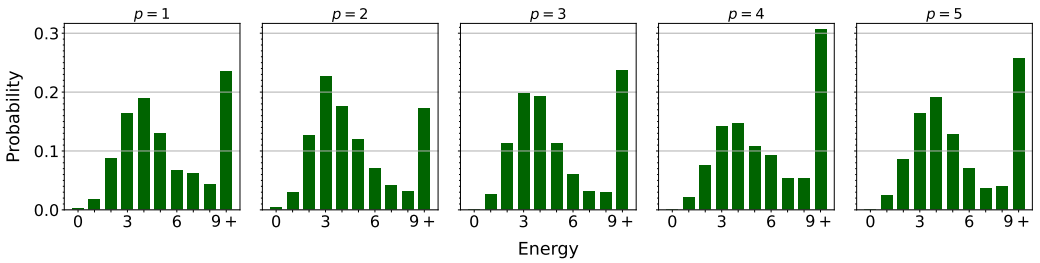


Figure 13. Probability of measuring a solution with given energy level, for different circuit lengths using QAOA algorithm.

6. Discussion

We were looking for the assignment of the appropriate speed and charging point to EMV traveling on the motorway so that no one has to wait for the charging process to start after reaching the station. We considered two models of vehicle charging points:

1. A set of independent chargers (probably some of them at the same station).
2. A set of charging stations with several identical chargers.

In the first case, we treat each charger separately and search for conflicts consisting in connecting more than one vehicle to the same charger at the same time. This is a more accurate approach to the problem under consideration. In the second case, we enable charging a number of vehicles in parallel at the same station. As described earlier, such an approach, despite the lower memory consumption, causes the rejection of acceptable solutions due to the detection of a greater number of conflicts (Figure 7). However, we use this approach because of the limited technological capabilities of quantum computers (low capacity) and the limited time of computing. Minimizing the number of variables and simplifying the conflict matrix allows finding feasible solutions for instances of the greater size.

In the graphs provided in Section 5, the parameter solution energy value should be interpreted as a value proportional to the number of unresolved conflicts at charging points. As a result of the experiments, the expected effect was observed—the more EMV, the more potential conflicts and the more difficult it is to find a feasible solution to the conflict avoidance problem. It can be assumed that in these cases, an acceptable solution does not exist. However, it should be taken into account that the tested algorithms should be considered as heuristics, which do not guarantee finding the sought solutions to the problem. Their great advantage, however, is the short computation time (evident for the QUBO case) and low energy consumption for computation.

At the current level of quantum technology development, larger instances of the problem can be solved on D-wave computers. It should be noted that in our experiments, we used BQM only—a purely quantum model of computation. The D-wave company provides another way of solution computation using hybrid (both quantum and classical computing). The constraint quadratic model (CQM) allows for defining inequality constraints, as well as many other utilities. What is more, this model type is required to use hybrid computing in which the problem is first divided into sub-problems and only then solved, piece by piece, on a purely quantum machine. This approach allowed us to reliably solve similar instances containing 40 EMVs (compared to BQM's 10). Unfortunately, as the computing time on D-wave machines is highly limited, we could only run a handful of experiments.

The problem we are discussing can be developed toward an optimization problem. We will notice that when using the discussed method of work, one can consider the problem, among others, of minimizing the travel time assuming one or more charges or minimizing the energy consumption of all tested vehicles. Another limitation that can be taken into account is the limitation of the power or energy available at specific charging stations, which would further reduce the number of vehicles charged at the same time.

7. Conclusions

In this paper, a problem of charging batteries of EMVs driving on a motorway was considered. The problem was formulated in terms of the deterministic scheduling theory. We looked for a feasible solution in which no EMV has to wait for starting the process of charging its battery. Two quantum computing approaches were proposed to attack the problem—gate-based approach (QAOA) and quantum annealing (QA). A computational experiment was designed and carried out in order to evaluate the efficiency of the proposed quantum algorithms. It can be seen that quantum technology is just developing, but its use is already becoming real and useful. With the assumed considerable limitations, the obtained results are still several times better than the random results.

The results show that the quality of the solution is strongly related to the number of variables (number of EMVs, charging stations, and driving modes). Due to the limitations of quantum machines and access to them, we managed to perform only one test instance with tangible results. The obtained result leaves us an open path for further experiments on this topic, as soon as there are more possibilities for the use of quantum computers.

Let us notice that although our model contains some simplifications (e.g., a discrete number of driving modes defined by available speeds), it can still be useful in practice. Decisions made by such a centralized scheduling module can achieve an advantage over a

set of decisions made autonomously by EMV drivers. It has all the information coming from monitoring systems on the motorway, as well as from the vehicles themselves. This may enable to synchronize and optimize the entire process of charging the fleet of EMVs which an individual driver is not capable of doing. Nowadays, there are already technical means sufficient to carry out the entire process. Additionally, despite the fact that the situation on a motorway is very dynamic and, generally, it would require online scheduling, the quantum computing power may be enough to apply the batch scheduling approach and solve the problem for a fleet of EMVs currently present on a considered motorway section. This approach will become practically more applicable in the near future when quantum computers become more powerful.

Future research can go in several directions. First of all, the considered problem can be generalized in many different ways. Possible extensions may include generalizations of the EMV model, motor way infrastructure, and/or charging process assumptions. In this paper, we only analyzed a decision (deconfliction) problem. On its basis, various optimization problems can be formulated in which different objective functions may be studied, e.g., minimization of the total (or mean) flow time, total waiting time, the number of waiting EMVs, and energy consumption. Additionally, from the computational point of view, various extensions of the experiment are possible, including solving bigger instances and comparing quantum approaches to some classical ones, e.g., involving local search metaheuristics.

Author Contributions: Conceptualization, R.R.; methodology, R.R., K.K. and G.W.; software, T.L., T.P. and M.S.; validation, R.R., J.J., K.K. and G.W.; formal analysis, M.S.; investigation, T.L. and T.P.; resources, J.J. and K.K.; data curation, T.L. and M.S.; writing—original draft preparation, all authors; writing—review and editing, all authors; visualization, T.P.; supervision, R.R.; project administration, J.J. and G.W.; funding acquisition, J.J. and K.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Poznan University of Technology, project number 0311/SBAD/0708.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

EMV	Electric Motor Vehicle
NISQ	Noisy Intermediate Scale Quantum
QPU	Quantum Processing Unit
QUBO	Quadratic Unconstrained Binary Optimization
QAOA	Quantum Approximate Optimization Algorithm
CM	Conflict Matrix
GCM	General Conflict Matrix
SCM	Station Conflict Matrix
BQM	Binary Quadratic Model
CQM	Constraint Quadratic Model

References

1. Wiśniewski, J.; Zagorzycka, K.; Kania, A.; Witkowski, Ł. Poland Drives E-Mobility. In Proceedings of the 17th International Conference on Project Management and Scheduling, Toulouse, France, 21–23 April 2022.
2. Lemański, T.; Różycki, R.; Waligóra, G.; Węglarz, J. Scheduling of battery charging tasks with limited common power source. In Proceedings of the 17th International Conference on Project Management and Scheduling, Toulouse, France, 21–23 April 2021.

3. Różycki, R.; Waligóra, G.; Węglarz, J. Scheduling battery charging jobs with linearly decreasing power demands to minimize the total time. In *Bulletin of the Polish Academy of Sciences: Technical Sciences*; Polska Akademia Nauk: Warsaw, Poland, 2020.
4. Różycki, R.; Lemański, T.; Józefowska, J. Scheduling UAV's on Autonomous Charging Station. *Mod. Technol. Enabling Saf. Secur. Uav Oper. Urban Airsp.* **2021**, *59*, 93.
5. Kim, H.; Shin, K.G. Scheduling of battery charge, discharge, and rest. In Proceedings of the 2009 30th IEEE Real-Time Systems Symposium, Washington, DC, USA, 1–4 December 2009; pp. 13–22.
6. Mitici, M.; Pereira, M.; Oliviero, F. Electric flight scheduling with battery-charging and battery-swapping opportunities. *Euro J. Transp. Logist.* **2022**, *11*, 100074. [CrossRef]
7. Tan, X.; Qu, G.; Sun, B.; Li, N.; Tsang, D.H. Optimal scheduling of battery charging station serving electric vehicles based on battery swapping. *IEEE Trans. Smart Grid* **2017**, *10*, 1372–1384. [CrossRef]
8. You, P.; Yang, Z.; Zhang, Y.; Low, S.; Sun, Y. Optimal charging schedule for a battery switching station serving electric buses. *IEEE Trans. Power Syst.* **2015**, *31*, 3473–3483. [CrossRef]
9. Schaden, B.; Jatschka, T.; Limmer, S.; Raidl, G.R. Smart Charging of Electric Vehicles Considering SOC-Dependent Maximum Charging Powers. *Energies* **2021**, *14*, 7755. [CrossRef]
10. Yang, S.N.; Cheng, W.S.; Hsu, Y.C.; Gan, C.H.; Lin, Y.B. Charge scheduling of electric vehicles in highways. *Math. Comput. Model.* **2013**, *57*, 2873–2882. [CrossRef]
11. del Razo, V.; Jacobsen, H.A. Smart charging schedules for highway travel with electric vehicles. *IEEE Trans. Transp. Electrif.* **2016**, *2*, 160–173. [CrossRef]
12. Ajagekar, A.; You, F. Quantum computing for energy systems optimization: Challenges and opportunities. *Energy* **2019**, *179*, 76–89. [CrossRef]
13. Faugler, J. *Big Data in der Energiewirtschaft: Warum Eon Quantencomputer Einsetzt*; Handelsblatt: Düsseldorf, Germany, 2020.
14. Xin, G.; Wang, P.; Jiao, Y. Noise-enhanced quantum annealing approach and its application in plug-in hybrid electric vehicle charging optimization. *Electron. Lett.* **2021**, *57*, 314–316. [CrossRef]
15. Kurowski, K.; Węglarz, J.; Subocz, M.; Różycki, R.; Waligóra, G. Hybrid quantum annealing heuristic method for solving job shop scheduling problem. In Proceedings of the International Conference on Computational Science, London, UK, 21–23 June 2022; Springer: Berlin/Heidelberg, Germany, 2020; pp. 502–515.
16. Ikeda, K.; Nakamura, Y.; Humble, T.S. Application of quantum annealing to nurse scheduling problem. *Sci. Rep.* **2019**, *9*, 12837. [CrossRef] [PubMed]
17. Błażewicz, J.; Ecker, K.; Pesch, E.; Schmidt, G.; Sterna, M.; Węglarz, J. *Handbook on Scheduling: From Theory to Practice*; Springer: Cham, Switzerland, 2019.
18. Kordmostafapour, A.; Rezaeian, J.; Mahdavi, I.; Yar Farjad, M. Scheduling unrelated parallel machine problem with multi-mode processing times and batch delivery cost. *OPSEARCH* **2022**, *59*, 1438–1470. [CrossRef]
19. Nielsen, M.A.; Chuang, I. *Quantum Computation and Quantum Information*; American Association of Physics Teachers: College Park, MD, USA, 2002; pp. 185–198.
20. Kitaev, A.Y. Quantum computations: Algorithms and error correction. *Russ. Math. Surv.* **1997**, *52*, 1191. [CrossRef]
21. Chow, J.; Dial, O.; Gambetta, J. IBM Quantum breaks the 100-qubit processor barrier. *Ibm Res. Blog* **2021**. Available online: https://research.ibm.com/blog/127-qubit-quantum-processor-eagle?lnk=ushpv18nf2 (accessed on 1 September 2022).
22. Farhi, E.; Goldstone, J.; Gutmann, S. A quantum approximate optimization algorithm. *arXiv* **2014**, arXiv:1411.4028.
23. Farhi, E.; Goldstone, J.; Gutmann, S.; Sipser, M. Quantum computation by adiabatic evolution. *arXiv* **2000**, arXiv:quant-ph/0001106.
24. Hadfield, S. On the representation of Boolean and real functions as Hamiltonians for quantum computing. *Acm Trans. Quantum Comput.* **2021**, *2*, 1–21. [CrossRef]
25. Wager, G.; Whale, J.; Braunl, T. Driving electric vehicles at highway speeds: The effect of higher driving speeds on energy consumption and driving range for electric vehicles in Australia. *Renew. Sustain. Energy Rev.* **2016**, *63*, 158–165. [CrossRef]
26. Schmidt, M.; Zmuda-Trzebiatowski, P.; Kiciński, M.; Sawicki, P.; Lasak, K. Multiple-Criteria-Based Electric Vehicle Charging Infrastructure Design Problem. *Energies* **2021**, *14*, 3214. [CrossRef]
27. Wykaz Parkingów i Mop—Generalna Dyrekcja Dróg Krajowych i Autostrad—Portal gov.pl. Available online: https://www.gov.pl/web/gddkia/wykaz-parkingow-i-mop (accessed on 1 September 2022).
28. D-Wave Leap Platform. Available online: https://docs.dwavesys.com/docs/latest/c_qpu_timing.html (accessed on 1 September 2022).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

Publication [P2]

Kurowski, K., Pecyna, T., Slysz, M., Różycki, R., Waligóra, G., Węglarz, J.: Application of quantum approximate optimization algorithm to job shop scheduling problem. European Journal of Operational Research. 310, 518–528 (2023). https://doi.org/10.1016/j.ejor.2023.03.013.

Ministry points / journal: 140

Contribution of authors:

- Krzysztof Kurowski:
  - Authorship of the idea underlying the paper.
  - Co-authorship of the text of publication (Sections: Introduction)
  - Review and editing of the paper
  - Results analysis
- Tomasz Pecyna:
  - Authorship of the idea of the quantum approach
  - Implementation of the quantum approach
  - Performing experiments, results analysis
  - Writing the text of publication
- Mateusz Slysz:
  - Co-authorship of the text of publication (Sections: Introduction)
  - Theoretical support and consulting on the quantum optimization approaches
- Rafał Różycki:
  - Co-authorship of the text of publication (Sections: Introduction, State of the art, Job shop scheduling problem)
  - Review and editing of the paper
- Grzegorz Waligóra:
  - Co-authorship of the text of publication (Sections: Introduction, State of the art, Job shop scheduling problem)
  - Review and editing of the paper
- Jan Węglarz:
  - Consulting on the computational complexity theory

European Journal of Operational Research 310 (2023) 518–528

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Discrete Optimization

Application of quantum approximate optimization algorithm to job shop scheduling problem

Krzysztof Kurowski[a,*], Tomasz Pecyna[a], Mateusz Slysz[a], Rafał Różycki[b], Grzegorz Waligóra[b], Jan Węglarz[b]

[a] Poznan Supercomputing and Networking Center, Poznan, IBCH PAS, Poland
[b] Poznan University of Technology, Institute of Computing Science, Poznan, Poland

ARTICLE INFO

Article history:
Received 6 October 2021
Accepted 8 March 2023
Available online 12 March 2023

Keywords:
Scheduling
Computing science
Heuristics
Job shop scheduling problem
Quantum approximate optimization algorithm

ABSTRACT

The Job Shop Scheduling Problem (JSSP) has always been considered as one of the most complex and industry essential scheduling problems. Optimizing the makespan of a given schedule generally involves using dedicated algorithms, local search strategies, or metaheuristics. These approaches, however, heavily rely on classical computational power, which is bounded by the physical limits of microcontrollers and power issues. Inspired by the promising results achieved for Quantum Annealing (QA) based approaches to solve JSSP instances, we propose a new approach that uses gate-model quantum architecture as an alternative to QA. We find that we can make use of the time-indexed JSSP instance representation to build a cost Hamiltonian, which can be embedded into Quantum Approximate Optimization Algorithm (QAOA) to find an optimal solution to a basic JSSP instance. We demonstrate the use of QAOA to solve the JSSP, and we evaluate its efficiency and accuracy for this problem from experimental results, as there is an increased urgency to demonstrate the applicability of quantum optimization algorithms. We also find that optimal variational parameters form patterns that can facilitate computation in bigger quantum circuits. Additionally, we compare the obtained noiseless simulation results of gate-model quantum circuits demonstrating the relationship between two evaluation criteria - makespan and energy. Finally, we analyze and present the overall performance of our approach with the increasing deadline and simulated depth of QAOA circuits.

© 2023 Poznan Supercomputing and Networking Center IBCH PAS. Published by Elsevier B.V.
This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/)

1. Introduction

The real value of quantum computers can be unlocked only through new applications, especially in operational research, where many difficult problems are of practical interest and hard to solve due to their computational complexity. Inspired by the unique features of the quantum realm, quantum computing promised from the early 1980s to deliver the ability to solve problems beyond classical computers' capability. Despite the impressive progress of quantum research, including recent studies and interesting attempts based on experimental quantum hardware, it is still unclear to many researchers how and to what extent they can benefit from quantum computations. They need to understand how to design, implement, and run a quantum computational experiment to solve a specific combinatorial optimization problem in a reliable and controlled way. It is worth introducing fundamental concepts and existing challenges before diving into our quantum algorithmic and experimental details.

There is a conceptual analogy and mathematical resemblance between the equations of an objective function used in an optimization problem and a *Hamiltonian* formalism. In a nutshell, the Hamiltonian, with its cost value, is an operator for the total energy of a system in quantum mechanics. Historically, it was realized back in the 1980s that simulating quantum dynamics to find the ground state energy of even a small molecule is far too complex for a classical computer as the computational power required to describe a quantum system scales exponentially with the number of components. Thus, the idea of using quantum instead of a classical computer to simulate other quantum systems evolved slowly after new concepts connecting quantum computation and ground states of many-body quantum systems were discussed in Feynman (1982). Additionally, the construction of a microscopic quantum mechanical Hamiltonian model of the computation processes represented by Turing machines was proposed in Benioff (1980). Over the next decade, there have been a lot of theoret-

* Corresponding author.
E-mail address: krzysztof.kurowski@man.poznan.pl (K. Kurowski).

https://doi.org/10.1016/j.ejor.2023.03.013
0377-2217/© 2023 Poznan Supercomputing and Networking Center IBCH PAS. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/)

ical attempts to demonstrate the potential of quantum mechanical computers and their computational properties, including but not limited to concepts of universal and inefficient quantum Turing machine presented initially in Deutsch (1985) and equivalence between quantum Turing machines and uniform quantum circuits in Yao (1993).

Although the tremendous theoretical results sparked a lot of interest among some researchers, many doubts about their practical significance and applications remained. One of the biggest technical challenges towards scalable and reliable quantum computing is decoherence phenomenon. Compared with classical computers, quantum computers are extremely susceptible to noise. Decoherence is the permanent adversary of quantum information processing since it destroys the fragile superpositions where information is stored. Lower decoherence can alleviate the error correction overhead and significantly reduce noise in quantum circuits and processing. Basically, the greater the influence of noise, the shorter the quantum algorithm that can be run before it suffers an error and outputs an incorrect result. Thus, stability and error correction issues remain an obstacle for unlocking quantum supremacy in solving real-world problems today. To describe the current state of the art in the fabrication of quantum circuits the concept of *Noisy Intermediate-Scale Quantum* (NISQ) era was introduced recently in Preskill (2018). It is clear that even today special attention must be paid to quantum error correction as NISQ devices are still not fault-tolerant and contain only a limited number (going into hundreds) of available basic units of quantum information - *qubits*. Thus, it is essential to identify a computing task or problem of interest to operational research that hopefully can be performed more efficiently, with better quality or more cost-effectively using quantum computers. Today, even without access to NISQ devices, advanced analysis and experimental tests of new quantum algorithms relevant for operational research can be performed thanks to quantum simulators widely available, as we demonstrate in this paper. Consequently, new quantum algorithms could be evaluated experimentally, adapted and tuned to constantly improved quantum NISQ devices, and we could have practical applications in the future. Last but not least, quantum computers promise to use much less energy and still vastly outperform supercomputers in the future. Modern classical supercomputers use between one to several megawatts of power on average. In contrast, existing NISQ devices use tens of kilowatts and generate almost no heat. Naturally, energy efficiency will depend on the concrete architecture and thus on available technological solutions. However, we may expect significant progress and a technological breakthrough in this relevant and economic aspect of computations.

The rest of this paper is organized as follows. In Section 2 we give a state of the art and brief review of the related works. In Section 3 we present the problem formulation. In Section 4 we formulate the Ising Hamiltonian representation applicable to the Quantum Approximate Optimization Algorithm (QAOA). Computational experiments and the obtained results are described in Section 5. Section 6 concludes the paper and shows some directions for future work.

2. State of the art

From a conceptual computational complexity perspective, the Deutsch-Jozsa algorithm was the first to show a separation between the quantum and classical difficulty of a problem (Deutsch & Jozsa, 1992). Then, a set of first truly and historically important quantum algorithms were proposed, in particular quantum polynomial-time algorithms for the discrete logarithm and integer factoring problems (Shor, 1994), the first algorithm to solve a promise problem exponentially faster than any classical algorithm (Simon, 1994), or the quantum search algorithm achieving quadratic speedup in unordered database search (Grover, 1996). Although we have some evidence of quantum algorithms performing better than their classical equivalents we are far from precisely establishing the true power of quantum computers. The quantum complexity theory was naturally investigated in Bernstein & Vazirani (1993), Yao (1993), and a new class of computational problems called '*Bounded error, Quantum, Polynomial time*' (BQP) was then introduced in Bernstein & Vazirani (1997). BQP consists of those decision problems that are solvable with bounded probability of error using a polynomial-size quantum circuit, and examples of problems belonging to this class can be the aforementioned integer factorization. We know so far that BQP is contained inside PSPACE class, which is the class of decision problems solvable by a Turing machine in polynomial space, and that it contains the BPP (problems which can be solved using randomized algorithms in polynomial time if bounded by probability error), hence the P class (Fortnow & Rogers, 1999). It is then determined that P $\subseteq$ BPP $\subseteq$ BQP $\subseteq$ PSPACE $\subseteq$ EXP. Since we know that P $\subset$ EXP and do not know which of the inclusion is stricter, and also that we do not know how BQP relates to NP, researchers suspect that there might be some problems outside NP that quantum computers can solve efficiently. Another motivation towards quantum computers is the Solovay Kitaev theorem (Kitaev, 1997) which states that an arbitrary single qubit gate may be approximated to some accuracy using polylogarithmic number of gates from a predefined universal discrete set, so we know that we can construct quantum circuits efficiently.

The next relevant step was the introduction of quantum fluctuations into the well-known simulated annealing process of optimization problems in Kadowaki & Nishimori (1998). Then, quantum computation by adiabatic evolution was proved in Farhi, Goldstone, Gutmann, & Sipser (2000), and suggested a novel quantum algorithm for solving the satisfiability problem and other combinatorial search problems. Additionally, adiabatic computation has been shown to be polynomially equivalent to conventional quantum computing in the quantum gate model (Aharonov et al., 2007). In principle, the concept of Quantum Annealing (QA) came from the well-known metaheuristic optimization technique called Simulated Annealing (SA), in which the space of admissible solutions to a given optimization problem is penetrated by temperature-dependent random movements. The cost function defines the total energy of the solution space, and the solution space can be efficiently explored thanks to thermal fluctuations. The basic concept of simulating annealing has been adopted and implemented in quantum hardware successfully when quantum fluctuations have replaced thermal fluctuations known from the classical SA approach (Boixo, Albash, Spedalieri, Chancellor, & Lidar, 2013; Humble et al., 2013). Consequently, we have been observing a rapid growth of QA approaches successfully used for solving optimization problems formulated in terms of finding ground states of classical Ising spin Hamiltonians (Lucas, 2014). Today's quantum annealing devices (e.g. D-Wave) can be used to solve small instances of combinatorial optimization problems, including the Job Shop Scheduling Problem (JSSP), as it has been demonstrated in Venturelli, Marchand, & Rojo (2016) and Kurowski, Węglarz, Subocz, Różycki, & Waligóra (2020).

One should note that state of the art quantum computers in the NISQ era can only be easily applied to some problems, but they perform pretty well for some computational problems. If encoded correctly, selected problems can be solved on gate-based quantum computers (e.g. IBM quantum processing unit) thanks to variational hybrid quantum-classical algorithms. In general, efficient variational hybrid quantum-classical approaches include two leading algorithms, namely the Variational Quantum Eigensolver (VQE) (Cerezo et al., 2021; Coveney & Highfield, 2020; Ralli, Love, Tranter, & Coveney, 2021) and Quantum Approximate Optimization Al-

gorithm (QAOA) (Farhi, Goldstone, & Gutmann, 2014). QAOA is a hybrid (quantum-classical) algorithm that approximates the value of the optimal solution of a binary optimization problem with its accuracy controlled by the hyperparameter p, which is a small positive integer. The cost function of the problem is mapped to a Hamiltonian represented by a quantum circuit with depth (length) dependent on p. The quantum circuit that implements the algorithm consists of unitary gates and is evaluated several times on a quantum device with respect to classically precomputed variable parameters, updated with every iteration.

QAOA has already been applied to a few well-known combinatorial optimization problems, such as Max-Cut (Crooks, 2018), Travelling Salesman Problem (Radzihovsky, Murphy, & Mason, 2019), and Graph Coloring (Tabi et al., 2020). The author of the first paper studied the performance of QAOA on the MaxCut problem, optimizing the quantum circuits on a classical computer using automatic differentiation and stochastic gradient descent (Crooks, 2018). It was demonstrated that it is possible to amortize the training cost by optimizing batches of problem instances. The paper shows that QAOA can exceed the performance of the classical polynomial time Goemans-Williamson algorithm with modest circuit depth and that performance with fixed circuit depth is insensitive to problem size. Moreover, MaxCut QAOA can be efficiently implemented on a gate-based quantum computer with limited qubit connectivity using a qubit swap network. The presented results support the prospects that QAOA will be an efficient method for solving complex combinatorial optimization problems on near-term quantum computers. In the second paper the authors implemented and investigated two versions of QAOA for the Travelling Salesman Problem (TSP) (Radzihovsky et al., 2019). They call them Hamiltonian cost implementation and mixer implementation, respectively. The authors showed that their mixer implementation successfully solves the TSP and reproduces, up to cyclic permutations, the solution found by the classical solver. They also found that the mixer approach requires fewer gates than the Hamiltonian cost implementation, whereas the cost Hamiltonian is generally faster than the mixer. The authors stated that their implementations provide a glimpse into what problems quantum computers can solve and the possibility of utilizing quantum supremacy. They concluded that although they could not compare the quantum algorithms against classical algorithms on real-world scale problems, this will be an exciting area of future research as quantum hardware continues to improve. Finally, the authors introduced a novel space-efficient quantum optimization algorithm for the Graph Coloring Problem (Tabi et al., 2020). Their circuits were deeper than the ones of the standard approach. However, through a series of investigations, the authors presented the performance gain of this method. They showed that the required circuit width to embed the colouring problem was exponentially reduced in the number of colours. Although the depth of a single QAOA layer was increased, the number of required layers and optimization iteration steps to reach the optimal solution also decreased. The authors state that the proposed method and comparative study can be extended to a benchmarking framework for such performance gain analyses. Furthermore, they concluded that analogous space-efficient embedding techniques could be used to improve upon other graph-related quantum optimization methods.

Although QAOA is proven to improve the solution quality with increased depth, it is hard to scale this regularity on real NISQ quantum hardware due to errors caused by qubit imperfections such as cross-talk or decoherence, as well as errors caused by faulty gates and measurements. The current references demonstrate that a notable quantum advantage seems likely to be observed on real quantum hardware once the noise decreases by two orders of magnitude (Stilck França & Garcia-Patron, 2021). For this reason, researchers limit themselves in running QAOA experiments only to small circuits (De Palma, Marvian, Rouzé, & França, 2022; Harrigan et al., 2021; Khumalo, Chieza, Prag, & Woolway, 2022; Magann et al., 2021) of size $p = 1$ to $p = 3$. Nonetheless, alternatives that seem more robust to noise have been introduced, e.g. the feedback-based technique discussed in Magann, Rudinger, Grace, & Sarovar (2022).

In general, two main approaches address the issue of noise and errors in existing NISQ devices. The short-term approach is quantum error mitigation techniques used to deal with errors as they occur. Error mitigation techniques operate mainly in the post-processing stage, and therefore, they have limited usage because their usability decreases with circuit length and complexity. The long-term answer is the quantum error correction approach with techniques that can fix errors during circuit execution. With error correction, there exists a concept of a logical qubit, which can be composed of physical qubits, typically in the range between 5 (simple error correction techniques) and 1000 (perfect logical qubit). For more information about quantum error mitigation and correction, we refer the reader to standard textbooks, e.g. Nielsen & Chuang (2002). As our research focused on experimental verification of QAOA using a quantum simulator, not real NISQ quantum hardware, detailed analysis is outside the scope of this paper.

Nevertheless, even if the size of considered problem instances is small for today, we have been observing rapid development of quantum technologies, new error correction techniques, and constant growth of the number and better quality qubits. It will result in the possibility of solving much larger problem instances in the near future. Consequently, it is relevant to develop and verify experimentally new quantum-based approaches, such as QAOA, to complex combinatorial problems today since they will be able to cope with bigger and bigger problem instances until they finally surpass classical algorithms and prove quantum supremacy. Some promising symptoms of that have already been shown in the abovementioned papers.

Thus, this paper is aimed to identify a possible adaptation of QAOA to scheduling problems by conducting a relatively simple but comprehensive series of experiments to pave the way for the scheduling application's development. Due to many practical applications of scheduling in advanced planning, decision support and computer systems, we have selected the JSSP benchmark. In this paper, we show a set of advanced analyses and guide the reader step by step, demonstrating possible opportunities and limits in quantum simulation environments using still significant classical computational power, which can be supported or replaced by improved NISQ devices in the future.

3. Job shop scheduling problem

3.1. Problem formulation

The Job Shop Scheduling Problem (JSSP) has been one of the most studied optimization problems over a few decades. In the problem a set of dedicated (i.e. different) machines is to perform tasks of jobs, i.e. each job is composed of an ordered list of tasks, from among which every task requires a specific machine for a known processing time. There exist several constraints imposed on jobs and machines: (i) tasks are nonpreemptable, (ii) tasks of different jobs are independent, (iii) each task can be performed on one machine at a time, and (iv) each machine can process only one job at a time. The problem is to minimize the makespan, i.e. the maximum completion time of all tasks. The JSSP belongs to the most intractable scheduling problems considered in the literature. It is NP-hard in the strong sense, and only a few particular special cases are efficiently solvable. There are several different formulations of the JSSP (see e.g., Błażewicz, Dror, & Węglarz, 1991 for a survey), whereas a problem instance, as well as a feasible solution,

can be represented by a disjunctive graph or its specialized representation the graph matrix (Błażewicz, Pesch, & Sterna, 2000). In order to solve JSSPs exact methods, heuristic and metaheuristic algorithms have been used over the years (Błażewicz, Domschke, & Pesch, 1996; Jain & Meeran, 1999). The exact approaches have almost entirely been based on branch-and-bound procedures. These, as it is known, rely very much on strong lower bounds in order to cut branches of the enumeration tree as early as possible. Lower bounds for the JSSP have been analyzed in e.g., Brucker & Jurisch (1993) and Carlier & Pinson (1994). As far as approximation algorithms are concerned, many approaches have used priority rules to order the tasks of jobs. For an extended summary and discussion see Haupt (1989). Various schedule generation schemes for the JSSP with sequence-dependent setup times have been analyzed in Artigues, Lopez, & Ayache (2005). One of the most powerful procedures used for the problem is the shifting bottleneck heuristic (Adams, Balas, & Zawack, 1988; Pezzella & Merelli, 2000). Also constraint propagation approach has been widely used to solve the problem (Dorndorf, Pesch, & Phan-Huy, 2002). From among local search methods a great variety of approaches have been applied to the JSSP (see Vaessens, Aarts, & Lenstra, 1996 for a survey). Many metaheuristic algorithms have also been proposed over the years, including simulated annealing, tabu search, genetic algorithms, ant colony optimization, variable depth search, and their hybrids (see Błażewicz et al., 2019 for an extensive review). From among them excellent results are presented in Balas & Vazacopoulos (1998), where a guided local search with shifting bottleneck has been proposed, and in Zhang, Li, Rao, & Guan (2008) in which a simulated annealing/tabu search hybrid is described. In order to compare the efficiency of algorithms various benchmark sets are being used. A review of the current state of bounds on benchmark instances of the JSSP is given in van Hoorn (2018).

In this paper we use the following formulation of the JSSP. There are J jobs $\mathcal{J} = \{j_1, \ldots, j_J\}$ and each of them has O_j operations (tasks) $\mathcal{O}_j = \{o_{j1} \to \cdots \to o_{jO_j}\}$ to be processed in predefined order. Each of these single job operations must be processed on a specified and distinct machine from a set of M machines, $\mathcal{M} = \{m_1, \ldots, m_M\}$, and only one operation can be processed by a machine at a given time. Note that $\forall j\ O_j \leq M$. The objective is to find the minimum makespan, i.e., the earliest completion time of the last running job.

3.2. Problem representation

To effectively apply QAOA to the JSSP we need to find a representation, so that the feasibility constraints can be defined as a sum of binary clauses. Initially, a generic method for the decision version of JSSP together with experimental tests was presented in Venturelli et al. (2016). We have proposed an extension of this method for the optimization version of JSSP. In addition, when comparing two feasible solutions, in our approach, we guaranteed that the one with a shorter makespan would also be of lower energy. Consequently, a new Hamiltonian function responsible for optimizing makespan was proposed, parametrized, and considered during experimental feasibility studies for solving a well-known reference JSSP benchmark (FT06) on the D-Wave 2000Q quantum annealer (Kurowski et al., 2020). We use the time-indexed instance representation and for each operation we assign a set of binary variables representing specific timestamps on which the operation can start, i.e.:

$$x_{k,t} = \begin{cases} 1: & \text{if operation } o_k \text{ starts at time } t \\ 0: & \text{otherwise.} \end{cases} \tag{1}$$

The time t is bounded by an arbitrary deadline T common for all jobs, and the index k is a running index representing a position of an operation in a list that concatenates all operations over all jobs:

$$\left[\underbrace{o_{11}, \ldots, o_{1O_1}}_{j_1}, \underbrace{o_{21}, \ldots o_{2O_2}}_{j_2} \ldots, \underbrace{o_{J1}, \ldots o_{JO_J}}_{j_J}\right] = \left[\underbrace{o_1, \ldots, o_{k_1}}_{j_1}, \underbrace{o_{k_1+1}, \ldots o_{k_2}}_{j_2} \ldots, \underbrace{o_{k_{J-1}+1}, \ldots o_{k_J}}_{j_J}\right]. \tag{2}$$

Note, that in general the deadline T has to be estimated e.g. using some heuristic algorithm. In this paper, however, we restrain ourselves to setting T as a makespan of a random, feasible solution.

Let us now define a set of constraints which assures that the solution is feasible and ends before T. Firstly, the operation must start once and only once, which is expressed by the following formula:

$$\sum_k \left(\sum_t x_{k,t} - 1\right)^2 = 0. \tag{3}$$

Secondly, there can be only one operation running at a given machine at any time, i.e.:

$$\sum_m \left(\sum_{k,t,k',t' \in R_m} x_{k,t} x_{k',t'}\right) = 0, \tag{4}$$

where R_m is a union of two sets $R_m = A_m \cup B_m$. A_m is a set that constraints operation $o_{k'}$ to start on a machine m if operation o_k is still running on the machine, and B_m is a set that constraints two operations from starting at the same time unless one of their times is 0:

$$\begin{aligned} A_m = \{&(k,t,k',t') : (k,k') \in I_m \times I_m, \\ &k \neq k', 0 \leq t, t' \leq T, 0 < t' - t < l_k\}, \\ B_m = \{&(k,t,k',t') : (k,k') \in I_m \times I_m, \\ &k < k', t' = t, l_k > 0, l_{k'} > 0\}. \end{aligned}$$

In this notation we denote l_k as the processing time of operation k and I_m as the set of all operations that have to be processed on the machine m.

The last constraint is defined so that the original order of the operations is kept for all the operations for every job in a given instance:

$$\sum_{n=1}^{J} \left(\sum_{\substack{k_{n-1}<k<k_n \\ t+l_k>t'}} x_{k,t} x_{k+1,t'}\right) = 0. \tag{5}$$

If we name h_1, h_2, h_3 as the constraint objectives (3), (4) (5) we get:

$$h_1(x) = \sum_k \left(\sum_t x_{k,t} - 1\right)^2, \tag{6}$$

$$h_2(x) = \sum_m \left(\sum_{k,t,k',t' \in R_m} x_{k,t} x_{k',t'}\right), \tag{7}$$

$$h_3(x) = \sum_{n=1}^{J} \left(\sum_{\substack{k_{n-1}<k<k_n \\ t+l_k>t'}} x_{k,t} x_{k+1,t'}\right), \tag{8}$$

where x is a vector of length R representing all possible variables $x_{k,t}$. If the values of all three objectives are equal 0, then all the

constraints are satisfied, i.e., there is a feasible schedule and the makespan of this schedule is less or equal to T.

The Job Shop Scheduling Problem is not only limited to finding a feasible schedule, but also optimizing its makespan. We can take advantage of the time-indexed representation by deriving an additional term that will put a penalty favouring any optimal schedule over any non-optimal schedule.

Suppose that for our J jobs in a JSSP instance an optimal schedule finishes at a time τ. Since any job finishes when its last operation $o_{k_1}, o_{k_2}, \ldots, o_{k_J}$ is complete, we can penalize only the completion time of the last jobs' operations. Each last operation is therefore given a penalty of the form $base^{\textit{last-operation-completion-time}}$. Let $t_{k_1}, t_{k_2}, \ldots, t_{k_J}$ be the last operations' completion times in the optimal schedule. Let us also choose the base as $J+1$. We can show now that any optimal schedule will be less penalized that any non-optimal schedule.

The penalty that will be given to any optimal schedule takes the form of:

$$\sum_{n=1}^{J}(J+1)^{t_{k_n}}, \quad t_{k_n} \leq \tau, \tag{9}$$

and the most penalized optimal schedule, i.e., the schedule in which all the last operations finish at time τ will be given the penalty:

$$\sum_{n=1}^{J}(J+1)^{\tau} = J(J+1)^{\tau}. \tag{10}$$

Comparing (9) and (10) we can easily see that, indeed:

$$\sum_{n=1}^{J}(J+1)^{t_{k_n}} \leq J(J+1)^{\tau}. \tag{11}$$

Moreover, from (10) we can also see that:

$$J(J+1)^{\tau} < (J+1)(J+1)^{\tau} = (J+1)^{\tau+1}, \tag{12}$$

which tells us that the most penalized optimal schedule is always less penalized than an operation that completes at a smallest non-optimal time $\tau+1$. If we denote by $t'_{k_1}, t'_{k_2}, \ldots, t'_{k_J}$ the last operations' completion times of this non-optimal schedule (i.e., $\exists n : t'_{k_n} = \tau+1$), we can enhance (12) and write:

$$J(J+1)^{\tau} < (J+1)^{\tau+1} < \sum_{n=1}^{J}(J+1)^{t'_{k_n}}, \tag{13}$$

which ends our proof, that the most penalized optimal schedule is always less penalized than any non-optimal schedule.

Taking advantage of the derived penalties we can define an additional objective:

$$h_4(x) = \sum_{n=1}^{J}(J+1)^{t_{k_n}}, \tag{14}$$

which has the lowest values if the vector x produces an optimal schedule.

4. Quantum computing for the JSSP

4.1. Quantum computing

Preliminary to describing our approach of using QAOA to solve JSSP, it is useful to draft some basic concepts of quantum computing as it is a relatively new field of computer science. In the following subsection we will describe only the necessary ideas, an interested reader is referred to more comprehensive handbooks, e.g. Nielsen & Chuang (2002)

A qubit is a name for any two-level quantum system. A typical representation of a qubit takes the form of a ket

$$|\psi\rangle = a|0\rangle + b|1\rangle \tag{15}$$

by which we mean that we expect the qubit to be in the base state $|0\rangle$ with probability a^2 and to be in the base state $|1\rangle$ with probability b^2. This phenomena is called superposition. The parameters a and b are probability amplitudes of the qubit and satisfy the normalization criterion $a^2+b^2=1$. A system composed of R qubits is represented by

$$a_0|00\ldots00\rangle + a_1|00\ldots01\rangle + \cdots + a_{2^R}|11\ldots11\rangle, \tag{16}$$

where we write $|00\ldots00\rangle$ as an abbreviated form of a tensor product $|0\rangle \otimes |0\rangle \otimes \cdots \otimes |0\rangle \otimes |0\rangle$ and a_r is the probability amplitude of a corresponding base state. A qubit (or a system composed of many qubits) is in a given and prepared state and quantum state preparation is an essential subroutine for quantum computing. The only way to acquire knowledge about the qubit or the quantum system (i.e. there are hundreds of qubits) is to perform a measurement. However, the measurement does not give us complete knowledge about the state because once the qubits are measured, they always collapse into one of their base states. If we can prepare the same state many times, we can alleviate this problem by performing the measurement many times hence estimating the amplitudes up to some error.

Another useful way of describing a quantum state, which origins from quantum mechanics where it describes the total energy of a system, is by specifying its Hamiltonian. In quantum computing, a Hamiltonian is often user-defined and acts as an operator for an expected value measurement for some desired state. The expectation value of a quantum system (often referred to as energy) is mathematically written as

$$E = \langle\psi|H|\psi\rangle. \tag{17}$$

The Hamiltonian H is usually composed of scalar-weighted Pauli gates (Pauli, 1927), which in this case are used as the operators of orthogonal measurements. By its definition, the Hamiltonian can aggregate any function which can be described as a sum of binary clauses (Hadfield, 2021). In many quantum optimization algorithms the strategy is to aggregate into the Hamiltonian functions of interest e.g. functions describing constraints or the cost function and evolve an initial quantum state such that the expected value of the Hamiltonian is minimal. The state evolution is done using so-called quantum gates. Physically they take various forms which depend on quantum computer architecture, but mathematically they can always be described as unitary, linear transformations.

4.2. Quantum approximate optimization algorithm

Having formulated the JSSP, we can now describe the Quantum Approximate Optimization Algorithm and show a straightforward and intuitive transformation which takes the already derived objectives to cost Hamiltonians needed by QAOA.

Looking at the QAOA in a general way, suppose a combinatorial optimization problem is given. If we can define the problem objective function as a sum of finite number of clauses $C_\alpha(x)$ where $x \in \{0,1\}^R$ is a binary string of length R,

$$C(x) = \sum_{\alpha} C_\alpha(x), \tag{18}$$

then we can use QAOA to find an approximate solution for the problem with a given probability. To this end, we need to convert the clauses into quantum Hamiltonians by replacing the binary variables x_r, $r \in \{1,2,\ldots,R\}$, with spin variables s_r,

$$x_r = \frac{1-s_r}{2}, \tag{19}$$

and promoting each spin variable s_r to a Pauli-Z matrix σ_r^z. As a result, we obtain a cost Hamiltonian

$$H_C = C(\sigma^z). \tag{20}$$

The QAOA algorithm alternately applies the cost Hamiltonian and a mixing Hamiltonian H_B to the equal superposition $|+\rangle^{\otimes R}$ state p times, thus achieving a final state ψ:

$$|\psi_p(\vec{\gamma}, \vec{\beta})\rangle = e^{-i\beta_p H_B} e^{-i\gamma_p H_C} \dots e^{-i\beta_1 H_B} e^{-i\gamma_1 H_C} |+\rangle^{\otimes R}, \tag{21}$$

where the variables γ and β play the role of variational parameters to be optimized by a classical algorithm, and the mixing Hamiltonian H_B usually takes the form of a sum of Pauli-X matrices (Farhi et al., 2014):

$$H_B = \sum_{r=1}^{R} \sigma_r^x. \tag{22}$$

The goal of QAOA is to find such optimal parameters γ^* and β^*, so that the expected value

$$F_p(\vec{\gamma}, \vec{\beta}) = \langle \psi_p(\vec{\gamma}, \vec{\beta}) | H_C | \psi_p(\vec{\gamma}, \vec{\beta}) \rangle \tag{23}$$

is maximized (or minimized). In our work we will look for the so-called ground state of the cost Hamiltonian, hence we will aim to minimize the expected value. The expected value of the cost Hamiltonian will be referred to as energy in the following sections.

4.3. QAOA For the JSSP

Let us take the objectives (6) to (8) and (14) we derived beforehand and, using the formula (19), replace all the binary variables from vector x with spin variables s. Subsequently, let us promote the spin variables to Pauli-Z matrices. We can now treat the objective function expressed by the formula:

$$H_C(\sigma^z) = h_1(\sigma^z) + h_2(\sigma^z) + h_3(\sigma^z) + h_4(\sigma^z) \tag{24}$$

as the cost Hamiltonian (20). This cost Hamiltonian can be then embedded into the QAOA to find the solution of the JSSP.

5. Experiments and results

Due to computational power limitations required for quantum simulations, the following analysis is conducted on a toy JSSP instance. Our algorithm was launched using the Atos myQLM quantum simulator framework, allowing users to run quantum noiseless simulations up to several tens of cubits depending on the classical computer performance. The basic instance consists of 3 jobs and each of them contains between 1 and 2 operations with processing times ranging from 1 to 2 units. The number of machines is 3. The makespan for the optimal schedule for this instance is $\tau^* = 3$. We present the instance on a disjunctive graph in Fig. 1.

5.1. Patterns in the parameter space

For a given QAOA depth we can always find such variational parameters in the non-convex space that lie near the global optimum and further optimization gives no significant advantage towards lower energy. Let us call them the optimal variational parameters.

The succeeding optimal variational QAOA parameter sequences $\beta^* = \beta_1^*, \dots, \beta_i^*, \dots, \beta_p^*$ and $\gamma^* = \gamma_1^*, \dots, \gamma_i^*, \dots, \gamma_p^*$ might form patterns, i.e. their values might increase monotonically with increasing i and also, that they should interpolate the space with increasing p as demonstrated in Zhou, Wang, Choi, Pichler, & Lukin (2020). These properties are used in so-called *educated guess strategy* which is known to significantly speed up the process of finding

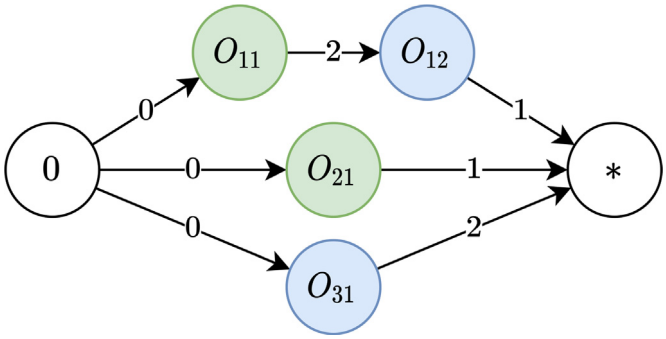


Fig. 1. The basic synthetic JSSP instance - the numbers on the directed edges represent processing tfimes of each operation, and the vertex color coding substitute undirected edges by marking operations that have to processed on the same machine.

optimal parameters (Vikstål et al., 2020; Zhou et al., 2020). The educated guess strategy will be discussed in the next subsection. To confirm that in the case of the JSSP's cost Hamiltonian these properties appear as well, we started our series of experiments by finding optimal parameters for an arbitrary p. Having found the optimal parameters we could plot landscapes of pairs (β_i, γ_i) ranging both β_i and γ_i from -2π to 2π. Fig. 2 shows the energy landscapes with monotonically traveling optimal pairs, which empirically confirms our assumptions.

5.2. Educated guess strategy

The educated guess interpolation strategy proposed in Zhou et al. (2020) assumes patterns in the parameter space. These patterns allow to take the optimal parameter sequences of depth p used by a quantum circuit of depth p, interpolate them to obtain parameter sequences of depth $p+1$ and pass them to the longer circuit of depth $p+1$. This is beneficial because (i) the longer the circuit is, the higher is the probability of obtaining a better solution (i.e., solution with low energy), and (ii) it takes more time to optimize parameters for longer circuits, so starting from initial points which are close to optimal ones can speed up the process of optimization.

In our implementation, we set p to a low value ($p = 3$) and then use multi-start strategy to initialize many random QAOA parameters and pass them as starting points to the local optimizer. The local optimizer was chosen to be the Constrained Optimization BY Linear Approximation (COBYLA) (Powell, 1994) algorithm, implemented in SciPy library, version 1.5.4 (Virtanen et al., 2020), and the function tolerance we set to 0.001. The number of starting points is empirically determined to be 500. After running the optimizer algorithm, we select several (5∼10) points, resulting in the lowest cost Hamiltonian energy. We interpolate them, as it was described in Zhou et al. (2020) to obtain new points of higher dimensionality. These points are then fed as parameters to the QAOA circuit of depth $p+1$ and optimized again by the COBYLA algorithm. We repeat the optimization-interpolation steps until a desired Hamiltonian energy is achieved on one of these points.

We start our series of experiments by defining the cost Hamiltonian as a sum of the feasibility constraints (6) to (8) only. This means that no penalty will be given on non-optimal schedules but the Hamiltonian will be less complex. In Fig. 3 we present a comprehensive visualisation of the behaviour of the variational parameters together with energy probabilities of the Hamiltonian. The data was collected using the aforementioned algorithm to the JSSP, choosing T to equal 4. We can see that both γ and β tend to monotonically increase their values in the domain of a single circuit and that they tend to interpolate the space in the domain of

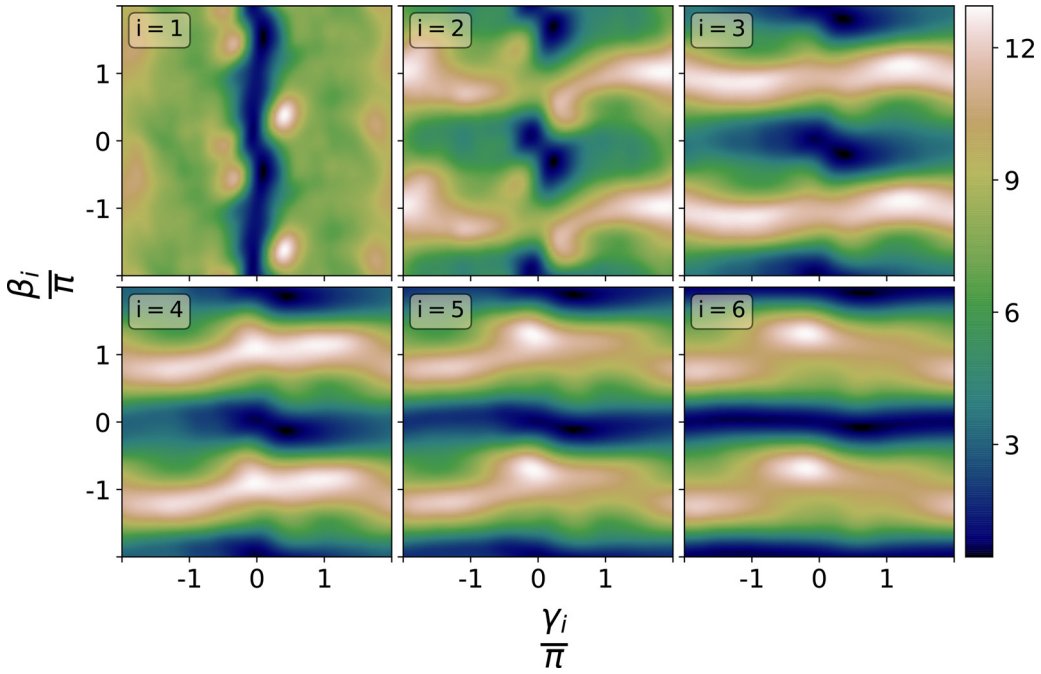


Fig. 2. Energy landscapes for the JSSP in a function of variational parameters β and γ for $p = 6$. On a single subplot we show only selected pairs of parameters (β_i, γ_i), while the remaining $(\beta_j, \gamma_j), i \neq j$ are set to constant, optimal values. We can observe repetition of the landscape energy values, which are the result of symmetries in variational parameters.

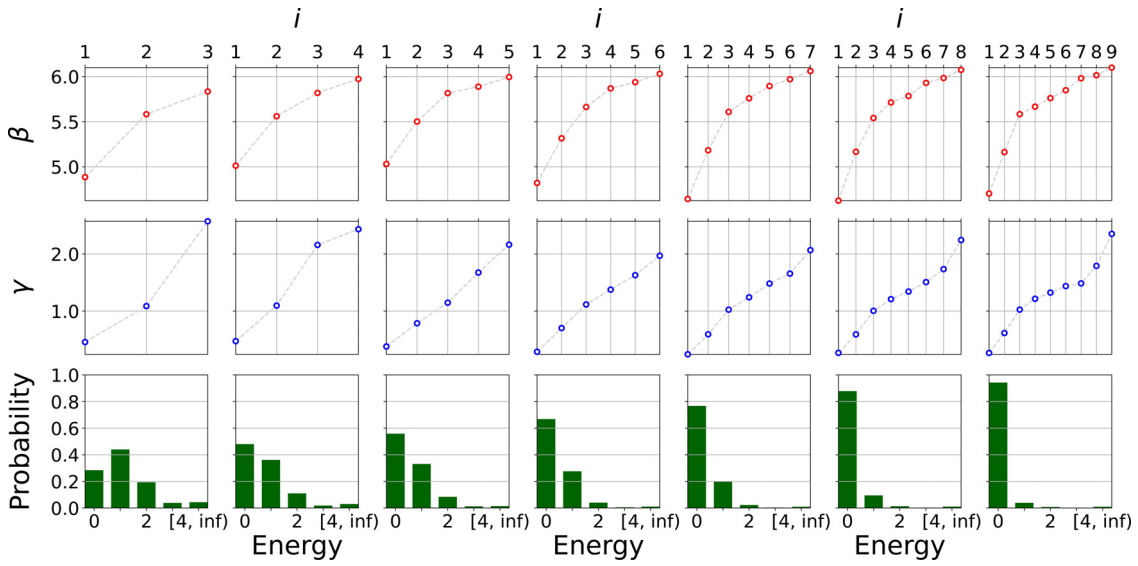


Fig. 3. Results for solving the toy JSSP instance using the educated-guess strategy with the Hamiltonian constructed of feasibility constraints only. The plots show 7 pairs of optimal parameters β and γ and corresponding energy probabilities of the cost Hamiltonian (solutions with energy greater than 4 are aggregated into one bar). Every next column represents one unit longer circuit than the previous one. Every next circuit was fed with previously found, interpolated and then optimized parameters (except the first circuit of depth $p = 3$ plotted on the first column, where the initial parameters were randomly chosen and optimized without interpolation).

increasing circuit depth p. We can also see that, indeed, the circuit depth affects the energy probabilities reaching over 90% chance of measuring the feasible solution when the depth $p = 9$.

We now proceed with the experiments by adding the fourth objective (14), which penalizes non-optimal schedules, to the Hamiltonian. Looking at the Fig. 4 we can see that the results are similar to the ones presented in Fig 3. Variational parameters make twin landscapes and they form similar patterns that interpolate the space with increasing circuit depth. Since now, the JSSP operations are penalized on their completion time, the minimal energy is almost always greater than 0 (unless all the processing times are 0, which is not the case in the instance). Comparing the energy probabilities of circuits with the same depth from both figures, we can see that they are roughly similar, e.g., the summed probability of measuring energies in range $[0.0, 0.5)$ and $[0.5, 1.0)$ in Fig. 3 with circuit of depth $p = 6$ is approximately equal to 0.64, which is comparable to probability of measuring energy value of 0 in Fig. 4 with circuit of depth $p = 6$. This leads to a conclusion that using the more complex Hamiltonian, seems to have little effect on the probabilities of measuring a solution with a low energy. This is good news since it also means that we can obtain low-makespan solution with no additional computational effort. The relation between energy and makespan will be the matter of the next subsection.

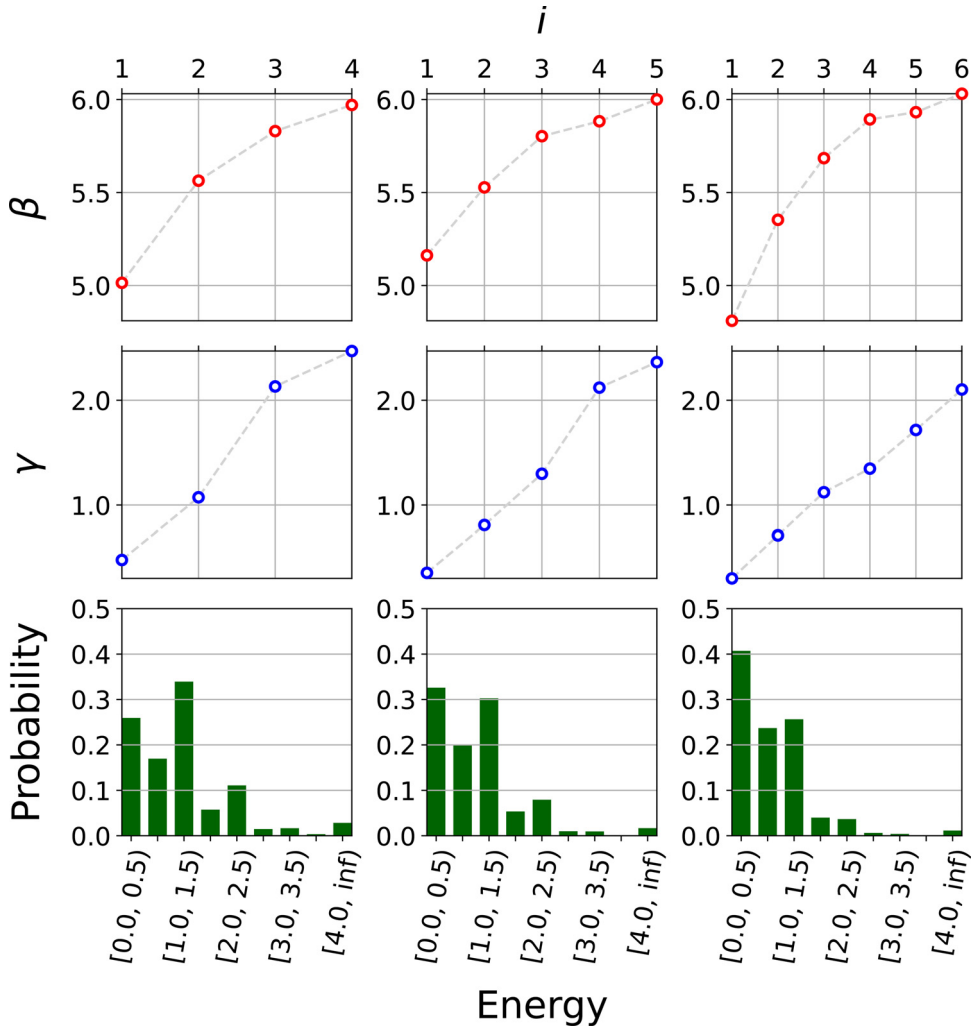


Fig. 4. Results for the JSSP instance using the educated-guess strategy with the Hamiltonian also penalizing solutions with high makespan. The subplots show 3 pairs of optimal parameters β and γ as well as corresponding energy probabilities of the cost Hamiltonian (solutions with close energy values are aggregated into single bins of range of 0.5).

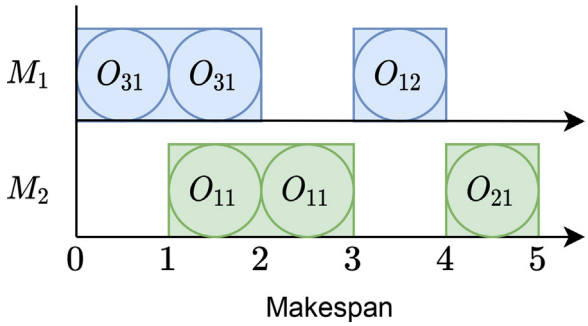


Fig. 5. A randomly sampled solution of the toy JSSP instance plotted on a Gantt chart. The operations O_{31} and O_{11} have processing time equal 2. The schedule generates energy 1.27 with the final makespan 5.

5.3. Performance analysis

We start the following analysis by randomly sampling a feasible solution for our toy instance. We plot the solution in Fig. 5. Its energy is approximately 1.27 and its makespan is 5. Based on this random solution, in the following series of experiments we set the deadline $T = 5$ and launch our energy-minimization algorithm to see how the mean energy decrease affects the probability of obtaining a solution with different (preferably lower) makespan. Afterwards, we lower the deadline to $T = 4$ in order to investigate the impact that hardening the constraints has on the probabilities of obtaining solutions with given energies.

We present the relation between energy and makespan in Fig. 6. The circuit used to produce data for this figure had depth $p = 5$. We can see a clear pattern that the lower the energy is, the higher is the probability of obtaining a feasible solution with a low makespan. Moreover, only the lowest range of energies correspond with the optimal solution (makespan 3). With circuit depth $p = 5$ approximately half of the solutions are infeasible, however. This is an expected behaviour that comes from the nature of quantum computing and can be reduced by iterative increasing the depth of the circuit, as was described in Section 5.2.

Interestingly enough, restricting the deadline to $T = 4$ resulted in rise in probability of measuring solutions with low energy, therefore lower makespan. This property might be advantageous when using any strategy based on repeated querying the circuit, e.g., starting with large T, measuring any feasible solution with makespan lower than T, and then setting this makespan value as T for the next iteration. Consequently, this would, again, accelerate the process of finding the optimal solution.

When considering performance we cannot underestimate the importance of time consumption, so in Fig. 7 we show the opti-

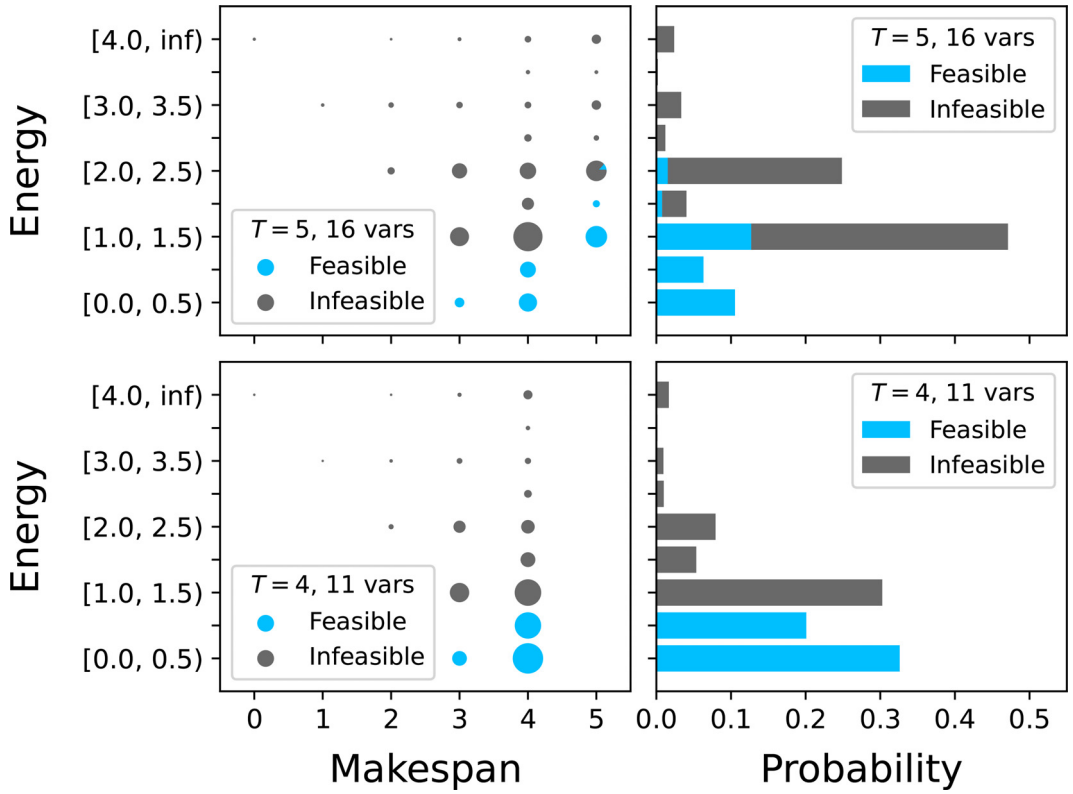


Fig. 6. Left hand-side: relationship between solution makespan and energy. Size of the markers are proportional to the probability of measuring a solution with given makespan and energy range. Right hand-side: marginal distribution of energy over the makespan. The top row present the results for deadline $T = 5$, while the bottom row: $T = 4$.

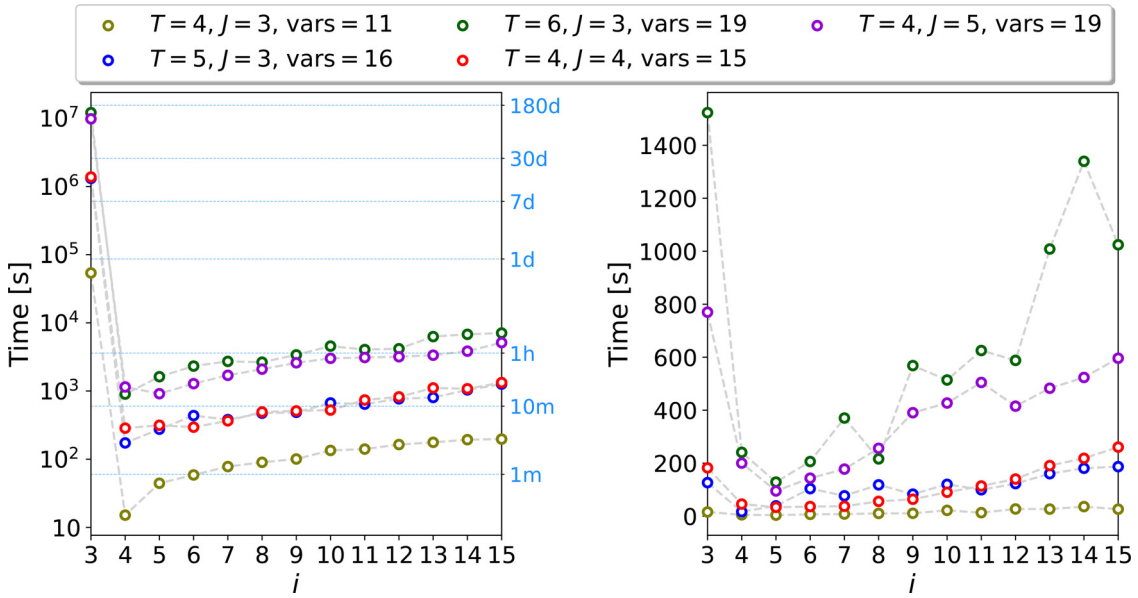


Fig. 7. Time needed for optimization of the variational parameters γ and β in the function of depth of simulated quantum gate-model circuit. The left hand-side shows the summed optimization time for all initial β and γ points per ith steps over all threads which were used. The right hand-side shows the optimization time of a single initial point, which resulted in the optimal solution.

mization time of the toy instance and its slight modifications (e.g. increased number of jobs). Then, we have enhanced the experiments up to 19 variables. We can observe an advantage of using the educated-guess strategy to solve the JSSP over a regular QAOA. Finding the optimal parameters is hard even for low-depth circuits, but if the optimal parameters are found and interpolated, the time needed to optimize them decreases significantly.

The second most apparent observation is the height of the overall duration time of the algorithm. For the toy instance, which is easily solvable by hand in several minutes by humans, it takes even days for a computer to solve, which might seem disappointing. Note, however, that in this paper, all the experiments have been made using a quantum simulator, meaning that all possible states (which number grows exponentially with the number of variables)

had to be processed sequentially on a classical computer. Thus, experimenting with only toy scheduling problem instances is necessary today when working with quantum simulators since quantum evolution requires the multiplication of quantum states by large matrices 2^{2n}. If we were to analyze with benchmarks from scientific literature, e.g. the FT06 benchmark for JSSP as we have demonstrated on a real quantum annealer, we would need at least hundreds of variables, which means that we would need RAM of around $2^{2 \cdot 100 \cdot 4}$ bytes on a classical computer to keep the quantum evolution matrices. Nevertheless, nowadays, several real gate-based NISQ devices with up to hundreds of qubits, e.g. recent IBM Q systems, are capable of handling this complexity with quantum computations.

We can also see, that the optimization time not only depends on the number of variables, but also depends on T or the instance parameters such as number of jobs. The reason for that is that QAOA's circuit depth is no longer dependent on the instance size n, but instead it is a function of the parameter p (Zhou et al., 2020). Even though these are not directly comparable, it is shown in Farhi & Harrow (2019) that it is enough to obtain quantum advantage in some cases.

6. Conclusions

This paper presented how to successfully implement a widely studied QAOA to solve a reference scheduling problem. We demonstrated how to efficiently apply QAOA to the well-known JSSP to find the optimal solution and pave the way for solving more complex scheduling problems in the future with more powerful and hybrid classic-quantum NISQ devices. Additionally, we investigated the behaviour of our energy-minimization algorithm in a series of experiments and demonstrated the relation between energy and makespan in achieving feasible and infeasible solutions. We discussed our experiences gained from real computational experiments and showed appropriate technical steps for quantum application developers interested in quantum simulation environments. All the presented results can be quickly adopted and extended by gate-model application developers, especially for initial testing and experimental verification of new quantum-based approaches for intensive quantum simulations. This paper can also be seen from the performance evaluation perspective of classical high-performance computer (HPC) systems and requirements for computing resources as new quantum-inspired and hybrid Quantum-HPC methodologies emerge.

Acknowledgments

This research has been partially supported by the statutory funds of Poznan University of Technology, Poznan Supercomputing and Networking Center affiliated to Institute of Bioorganic Chemistry Polish Academy of Sciences, and grant QATM, DOB-SZAFIR/01/B/023/01/2021. The calculations were performed at the Poznan Supercomputing and Networking Center. We express our gratitude to three anonymous referees whose valuable comments helped us significantly improve the quality of our paper.

References

Adams, J., Balas, E., & Zawack, D. (1988). The shifting bottleneck procedure for job shop scheduling. *Management Science, 34*(3), 391–401.

Aharonov, D., Van Dam, W., Kempe, J., Landau, Z., Lloyd, S., & Regev, O. (2007). Adiabatic quantum computation is equivalent to standard quantum computation. *SIAM Journal on Computing, 37*(1), 166–194. https://doi.org/10.1137/S0097539705447323.

Artigues, C., Lopez, P., & Ayache, P. (2005). Schedule generation schemes for the job-shop problem with sequence-dependent setup times: Dominance properties and computational analysis. *Annals of Operations Research, 138*, 21–52.

Balas, E., & Vazacopoulos, A. (1998). Guided local search with shifting bottleneck for job shop scheduling. *Management Science, 44*(2), 262–275.

Benioff, P. (1980). The computer as a physical system: A microscopic quantum mechanical hamiltonian model of computers as represented by turing machines. *Journal of Statistical Physics, 22*(5), 563–591. https://doi.org/10.1007/BF01011339.

Bernstein, E., & Vazirani, U. (1993). Quantum complexity theory. In *Proceedings of the twenty-fifth annual acm symposium on theory of computing*. In *STOC '93* (pp. 11–20). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/167088.167097.

Bernstein, E., & Vazirani, U. (1997). Quantum complexity theory. *SIAM Journal on Computing, 26*(5), 1411–1473.

Błażewicz, J., Domschke, W., & Pesch, E. (1996). The job shop scheduling problem: Conventional and new solution techniques. *European Journal of Operational Research, 93*(1), 1–33.

Błażewicz, J., Dror, M., & Węglarz, J. (1991). Mathematical programming formulations for machine scheduling: A survey. *European Journal of Operational Research, 51*(3), 283–300.

Błażewicz, J., Ecker, K., Pesch, E., Schmidt, G., Sterna, M., & Węglarz, J. (2019). *Handbook on scheduling: From theory to practice*. Cham: Springer.

Błażewicz, J., Pesch, E., & Sterna, M. (2000). The disjunctive graph machine representation of the job shop problem. *European Journal of Operational Research, 127*(2), 317–331.

Boixo, S., Albash, T., Spedalieri, F., Chancellor, N., & Lidar, D. (2013). Experimental signature of programmable quantum annealing. *Nature Communications, 4*, 2067.

Brucker, P., & Jurisch, B. (1993). A new lower bound for the job shop scheduling problem. *European Journal of Operational Research, 64*(2), 156–167.

Carlier, P., & Pinson, E. (1994). Adjustment of heads and tails for the job shop problem. *European Journal of Operational Research, 78*(2), 146–161.

Cerezo, M., Arrasmith, A., Babbush, R., Benjamin, S. C., Endo, S., Fujii, K., … Coles, P. J. (2021). Variational quantum algorithms. *ArXiv* abs/2012.09265.

Coveney, P. V., & Highfield, R. R. (2020). From digital hype to analogue reality: Universal simulation beyond the quantum and exascale eras. *Journal of Computational Science, 46*, 101093. https://doi.org/10.1016/j.jocs.2020.101093. https://www.sciencedirect.com/science/article/pii/S1877750320300545

Crooks, G. E. (2018). Performance of the quantum approximate optimization algorithm on the maximum cut problem. *arXiv preprint* arXiv:1811.08419.

De Palma, G., Marvian, M., Rouzé, C., & França, D. S. (2022). Limitations of variational quantum algorithms: Aquantum optimal transport approach. *arXiv preprint* arXiv:2204.03455.

Deutsch, D. (1985). Quantum theory as a universal physical theory. *International Journal of Theoretical Physics, 24*(1), 1–41. https://doi.org/10.1007/BF00670071.

Deutsch, D., & Jozsa, R. (1992). Rapid solution of problems by quantum computation. *Proc. R. Soc. Lond. A, 439*, 553–558. https://doi.org/10.1098/rspa.1992.0167.

Dorndorf, U., Pesch, E., & Phan-Huy, T. (2002). Constraint propagation and problem decomposition: A preprocessing procedure for the job shop problem. *Annals of Operations Research, 115*, 125–145.

Farhi, E., Goldstone, J., & Gutmann, S. (2014). A quantum approximate optimization algorithm. *arXiv:1411.4028: Quantum Physics*.

Farhi, E., Goldstone, J., Gutmann, S., & Sipser, M. (2000). Quantum computation by adiabatic evolution. https://arxiv.org/abs/quant-ph/0001106. 10.48550/ARXIV.QUANT-PH/0001106.

Farhi, E., & Harrow, A. W. (2019). Quantum supremacy through the quantum approximate optimization algorithm. *arXiv:1602.07674: Quantum Physics*.

Feynman, R. P. (1982). Simulating physics with computers. *International Journal of Theoretical Physics, 21*(6), 467–488. https://doi.org/10.1007/BF02650179.

Fortnow, L., & Rogers, J. (1999). Complexity limitations on quantum computation. *Journal of Computer and System Sciences, 59*(2), 240–252.

Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. https://arxiv.org/abs/quant-ph/9605043. doi:10.48550/ARXIV.QUANT-PH/9605043.

Hadfield, S. (2021). On the representation of boolean and real functions as hamiltonians for quantum computing. *ACM Transactions on Quantum Computing, 2*(4), 1–21.

Harrigan, M. P., Sung, K. J., Neeley, M., Satzinger, K. J., Arute, F., Arya, K., … Boixo, S., et al., (2021). Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. *Nature Physics, 17*(3), 332–336.

Haupt, R. (1989). A survey of priority-rule based scheduling. *OR Spektrum, 11*(1), 3–16.

van Hoorn, J. (2018). The current state of bounds on benchmark instances of the job-shop scheduling problem. *Journal of Scheduling, 21*(1), 127–128.

Humble, T., McCaskey, A., Bennink, R., Billings, J., D'Azevedo, E., Sullivan, B. D., … Seddiqi, H. (2013). An integrated programming and development environment for adiabatic quantum optimization. *Computational Science & Discovery, 7*, 015006.

Jain, A., & Meeran, S. (1999). Deterministic job shop scheduling: Past, present and future. *European Journal of Operational Research, 113*(2), 390–434.

Kadowaki, T., & Nishimori, H. (1998). Quantum annealing in the transverse ising model. *Physical Review E, 58*(5), 5355–5363. https://doi.org/10.1103/physreve.58.5355.

Khumalo, M. T., Chieza, H. A., Prag, K., & Woolway, M. (2022). An investigation of ibm quantum computing device performance on combinatorial optimisation problems. *Neural Computing and Applications*, 1–16.

Kitaev, A. Y. (1997). Quantum computations: Algorithms and error correction. *Russian Mathematical Surveys, 52*(6), 1191.

Kurowski, K., Węglarz, J., Subocz, M., Różycki, R., & Waligóra, G. (2020). Hybrid quantum annealing heuristic method for solving job shop scheduling problem. In V. V. Krzhizhanovskaya, G. Závodszky, M. H. Lees, J. J. Dongarra, P. M. A. Sloot, S. Brissos, & J. Teixeira (Eds.), *Computational science – iccs 2020* (pp. 502–515). Cham: Springer International Publishing.

Lucas, A. (2014). Ising formulations of many NP problems. *Frontiers in Physics, 2*, 5. https://doi.org/10.3389/fphy.2014.00005. https://www.frontiersin.org/article/10.3389/fphy.2014.00005.

Magann, A. B., Arenz, C., Grace, M. D., Ho, T.-S., Kosut, R. L., McClean, J. R., … Sarovar, M. (2021). From pulses to circuits and back again: A quantum optimal control perspective on variational quantum algorithms. *PRX Quantum, 2*(1), 010101.

Magann, A. B., Rudinger, K. M., Grace, M. D., & Sarovar, M. (2022). Lyapunov-control-inspired strategies for quantum combinatorial optimization. *Physical Review A, 106*(6), 062414.

Nielsen, M. A., & Chuang, I. (2002). Quantum computation and quantum information.

Pauli, W. (1927). Zur quantenmechanik des magnetischen elektrons. *Zeitschrift für Physik A Hadrons and Nuclei, 43*(9), 601–623.

Pezzella, E., & Merelli, E. (2000). Tabu search method guided by shifting bottleneck for the job shop scheduling problem. *European Journal of Operational Research, 120*(2), 297–310.

Powell, M. J. (1994). A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in optimization and numerical analysis* (pp. 51–67). Springer.

Preskill, J. (2018). Quantum computing in the NISQ era and beyond. *Quantum, 2*, 79. https://doi.org/10.22331/q-2018-08-06-79.

Radzihovsky, M., Murphy, J., & Mason, S. (2019). A qaoa solution to the traveling salesman problem using pyquil. https://cs269q.stanford.edu/projects2019/radzihovsky_murphy_swofford_Y.pdf.

Ralli, A., Love, P. J., Tranter, A., & Coveney, P. V. (2021). Implementation of measurement reduction for the variational quantum eigensolver. *Physical Review Research, 3*(3), 033195. https://doi.org/10.1103/PhysRevResearch.3.033195.

Shor, P. (1994). Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science* (pp. 124–134). https://doi.org/10.1109/SFCS.1994.365700.

Simon, D. R. (1994). On the power of quantum computation. In *Proceedings of the 35th annual symposium on foundations of computer science*. In *SFCS '94* (pp. 116–123). USA: IEEE Computer Society. https://doi.org/10.1109/SFCS.1994.365701.

Stilck França, D., & Garcia-Patron, R. (2021). Limitations of optimization algorithms on noisy quantum devices. *Nature Physics, 17*(11), 1221–1227.

Tabi, Z., El-Safty, K. H., Kallus, Z., H'aga, P., Kozsik, T., Glos, A., & Zimbor'as, Z. (2020). Quantum optimization for the graph coloring problem with space-efficient embedding. *2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 56–62.

Vaessens, R., Aarts, E., & Lenstra, J. (1996). Job shop scheduling by local search. *INFORMS Journal of Computing, 8*(3), 302–317.

Venturelli, D., Marchand, D. J. J., & Rojo, G. (2016). Quantum annealing implementation of job-shop scheduling. *arXiv:1506.08479: Quantum Physics*.

Vikstål, P., Grönkvist, M., Svensson, M., Andersson, M., Johansson, G., & Ferrini, G. (2020). Applying the quantum approximate optimization algorithm to the tail-assignment problem. *Physical Review Applied, 14*(3), 034009. https://doi.org/10.1103/PhysRevApplied.14.034009.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., … SciPy 1. 0 Contributors (2020). Scipy 1.0: Fundamental algorithms for scientific computing in python. *Nature Methods, 17*, 261–272. https://doi.org/10.1038/s41592-019-0686-2.

Yao, A. C.-C. (1993). Quantum circuit complexity. In *Proceedings of 1993 ieee 34th annual foundations of computer science*. In *Proceedings of 1993 IEEE 34th Annual Foundations of Computer Science* (pp. 352–361). https://doi.org/10.1109/SFCS.1993.366852.

Zhang, C., Li, P., Rao, Y., & Guan, Z. (2008). A very fast TS/SA algorithm for the job shop scheduling problem. *Computers & Operations Research, 35*(1), 282–294.

Zhou, L., Wang, S.-T., Choi, S., Pichler, H., & Lukin, M. D. (2020). Quantum approximate optimization algorithm: performance, mechanism, and implementation on near-term devices. *Physical Review X, 10*(2), 021067. https://doi.org/10.1103/PhysRevX.10.021067.

Publication [P3]

Pecyna, T., Kurowski, K., Rózycki, R., Waligóra, G., Węglarz, J.: Quantum variational algorithms for the aircraft deconfliction problem. In: Lecture notes in computer science. pp. 307–320 (2024). https://doi.org/10.1007/978-3-031-63778-0_22.

Ministry points / conference: 140

Contribution of authors:

- Tomasz Pecyna:
  - Co-authorship of the idea underlying the paper.
  - Authorship of the idea of the quantum approach
  - Implementation of the quantum approach
  - Performing experiments, results analysis
  - Writing the text of publication
- Krzysztof Kurowski:
  - Co-authorship of the idea underlying the paper
  - Results analysis
- Rafał Różycki:
  - Co-authorship of the idea underlying the paper.
  - Resuts analysis
  - Review and editing of the paper
- Grzegorz Waligóra:
  - Co-authorship of the idea underlying the paper.
- Jan Węglarz:
  - Consulting on the mathematical programming approaches

Quantum variational algorithms for the aircraft deconfliction problem

Tomasz Pecyna[1][0000−0001−7473−3360], Krzysztof Kurowski[1][0000−0002−4478−6119], Rafał Różycki[2][0000−0002−2973−8266], Grzegorz Waligóra[2][0000−0003−2108−1113], and Jan Węglarz[2][0000−0002−2237−3479]

[1] Poznań Supercomputing and Networking Center, IBCH PAS
{tpecyna,krzysztof.kurowski}@man.poznan.pl
[2] Poznań University of Technology
Institute of Computing Science
Poznan, Poland
{rafal.rozycki,grzegorz.waligora,jan.weglarz}@cs.put.poznan.pl

Abstract. Tactical deconfliction problem involves resolving conflicts between aircraft to ensure safety while maintaining efficient trajectories. Several techniques exist to safely adjust aircraft parameters such as speed, heading angle, or flight level, with many relying on mixed-integer linear or nonlinear programming. These techniques, however, often encounter challenges in real-world applications due to computational complexity and scalability issues. This paper proposes a new quantum approach that applies the Quantum Approximate Optimization Algorithm (QAOA) and the Quantum Alternating Operator Ansatz (QAOAnsatz) to address the aircraft deconfliction problem. We present a formula for designing quantum Hamiltonians capable of handling a broad range of discretized maneuvers, with the aim of minimizing changes to original flight schedules while safely resolving conflicts. Our experiments show that a higher number of aircraft poses fewer challenges than a larger number of maneuvers. Additionally, we benchmark the newest IBM quantum processor and show that it successfully solves four out of five instances considered. Finally, we demonstrate that incorporating hard constraints into the mixer Hamiltonian makes QAOAnsatz superior to QAOA. These findings suggest quantum algorithms could be a valuable algorithmic candidate for addressing complex optimization problems in various domains, with implications for enhancing operational efficiency and safety in aviation and other sectors.

Keywords: Tactical Aircraft Deconfliction · Quantum Approximate Optimization Algorithm · Quantum Alternating Operator Ansatz.

1 Introduction

The global COVID-19 pandemic was not enough to stop the long-term trend of increasing demand for aviation services. According to Airports Council International, in 2023 the number of passengers reached almost 95% of the levels from

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

2019, and projections indicate a surpassing of the 2019 level in 2024 [1]. Along with this trend, problems with airspace congestion are returning, and the demand for specialized algorithms dealing with airspace management comes back, one of the problems being the tactical aircraft deconfliction.

In literature, aircraft deconfliction, also known as a conflict detection and resolution problem, refers to the natural and common challenge of ensuring appropriate and safe separation among aircraft operating in the same controlled airspace. The problem arises due to the limited airspace and the need to accommodate multiple aircraft at different directions, altitudes, speeds, and planned maneuvers. The problem has been a subject of interest among many researchers within the community. Despite extensive exploration of conflict detection and resolution, numerous models struggled to sufficiently address the challenges of considered problem, as noted in a seminal work by Kuchar and Yang [16]. Then, the work by Pallottino et al. [21] gained much community attention by introducing the velocity change model, which utilizes mixed-integer linear programming (MILP) to allow real-time maneuvering to resolve aircraft conflicts. This approach was further refined by Alonso-Ayuso et al. [3], who incorporated altitude changes, weather conditions and trajectory recovery into the model while maintaining real-time capabilities.

In a separate study [27], Vela et al. concentrated on addressing the problem of future conflicts, which could occur within a timeframe ranging from 15 to 45 minutes, to minimize fuel costs. They reported achieving near-optimal solutions using the MILP approach, incorporating control over both velocity and altitude. Furthermore, Omer [20] observed that air traffic controllers and aircraft pilots do not favor all velocity, heading, and altitude changes. Consequently, he suggested a discretization approach to facilitate easier handling by human operators, resulting in a minor increase in fuel consumption, amounting to a few kilograms.

Instead of employing MILP, some researchers have proposed using nonlinear programming to address the issue of aircraft deconfliction. In their study [7], Cafieri and Durand utilized Mixed Integer Nonlinear Programming (MINLP) as a natural choice to model separation conditions, addressing the problem using only velocity change. The study conducted by Alonso-Ayuso et al. [4] also applied MINLP formulation to solve the deconfliction problem via turn changes. One notable work that builds upon these two approaches and combines them was conducted by Cafieri and Omheni [8]. They suggest initially resolving the problem by adjusting heading angles and subsequently using this solution as a preprocessing step for modifying velocities.

Various other studies have explored the deconfliction problem, considering factors such as stochasticity and three-dimensional space [17], or employing a new method such as bilevel programming [9]. For an in-depth review of research on deconfliction over the past two decades, one should refer to [22].

Given the recent advancements in quantum computing and still persistent challenges in the broad domain of air traffic management, it is not surprising that researchers have been exploring alternative approaches. The initial study

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

that focused on the application of quantum computers in aviation was conducted by Stollenwerk et al. [25], who proposed a method to solve flight-gate assignment problem using the D-Wave 2000Q quantum annealer. Using the same device, Stollenwerk et al. [26] addressed the strategic aircraft deconfliction problem by incorporating takeoff delays into wind-optimal trajectories. Additionally, they outlined a simplified model for trajectory modifications proposing pairwise exclusive avoidance or introducing delays between two consecutive conflicts. The D-Wave 2000Q quantum annealer was also used to solve the Tail Assignment Problem [12] in a study presented by Martins et al. [18]. The problem had been addressed also thanks to classical simulation of a universal quantum computer in [28]. Real gate-based quantum hardware, however, was employed to successfully solve only toy instances of flight-gate assignment in [19, 10].

In this paper, we introduce a novel approach to address the tactical aircraft deconfliction problem using gate-base quantum computers. Inspired by the ideas presented in [20], we advocate for conflict resolution through discretized maneuvers. Our main contributions include designing a proper cost Hamiltonian for the Quantum Approximate Optimization Algorithm coupled with the effective relocation of a subset of hard constraints into the mixer Hamiltonian of the Quantum Alternating Operator Ansatz. Furthermore, we establish a connection with our previous research by benchmarking our approach against a widely-used circle problem dataset published by Rey and Hijazi [23], which has been downscaled to align with the capacity of current quantum machines.

The paper is organized as follows. In Section 2, we formulate the problem, both classically and in quantum terms. In Section 3, we show how to use our formulation with existing quantum algorithms. In Section 4, we describe the results, and conclude the paper with future work in Section 5.

2 Problem Representation and assumptions

Let us assume that during the flight, an aircraft must maintain a minimum separation of 5 nautical miles horizontally and 1000 feet vertically from other aircraft, where a nautical mile equals 1852 meters and a foot equals 30.48 cm. A conflict between two aircraft arises when a pair of aircraft violates at least one of these constraints. If a particular conflict is detected and resolved within five to thirty minutes, then we consider the tactical deconfliction. [22]. We further assume that aircraft motion can be described by a sequence of line segments, maintaining a constant speed within each segment and allowing instantaneous speed changes at the beginning of each segment.

2.1 Classical formulation

We present a graphical summary of our approach to the deconfliction problem in Figure 1. The diagram illustrates the key components of our methodology, including the set of proposed maneuvers and the conflict matrix, which is introduced mathematically later in this subsection.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

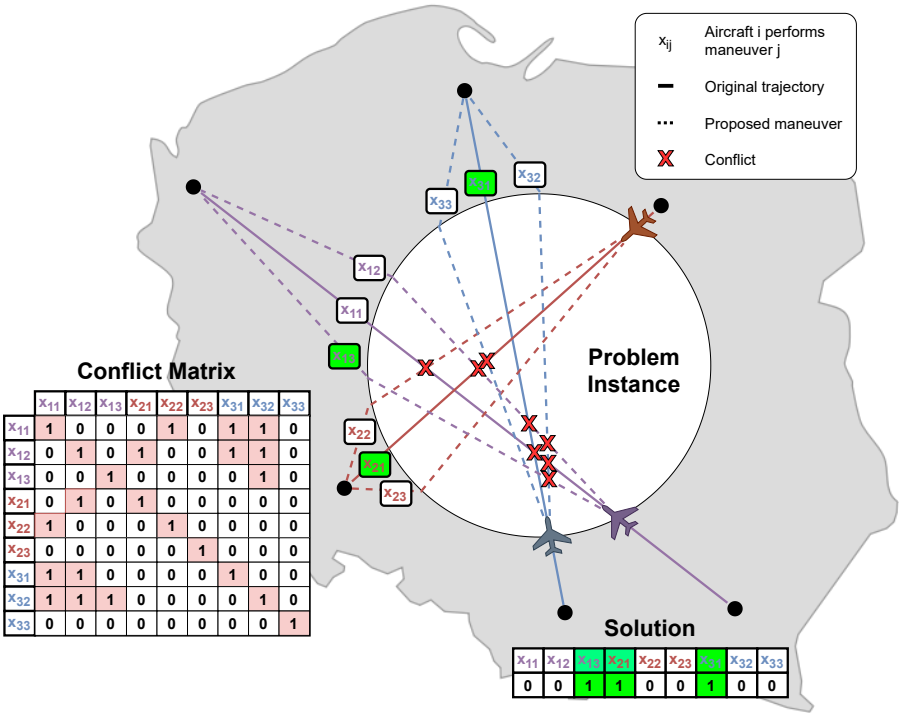


Fig. 1. Diagram summarizing our approach to the deconfliction problem. Initially, three aircraft are in conflict. After proposing 2 additional maneuvers (totaling 3 maneuvers), one feasible solution is proposed: aircraft 1 maneuver 3, aircraft 2 maneuver 1, aircraft 3 maneuver 1. After conflicts are resolved, aircraft may return to their original destinations, which, however, is beyond the scope of our approach.

Given a set of n aircraft with their respective positions, heading angles, speeds, and flight levels, our approach begins by proposing a set of discretized maneuvers for each aircraft. Maneuvers could be of various kinds, including heading angle change, speed change, or flight level change. For simplicity, we assume that each aircraft can perform m maneuvers, although the actual number may vary for an aircraft depending on specific flight requirements. To keep track of these maneuvers let us introduce a set of the following binary variables:

$$X = \{x_{ij} : \ i = 1 \ldots, n, \ j = 1, \ldots, m, \ x_{ij} \in \{0, 1\}\}. \tag{1}$$

If the variable x_{ij} is assigned the value 1 it indicates that the aircraft i performs maneuver j, whereas a value of 0 indicates the opposite. In this work, we assume that maneuvers are disjoint for an aircraft, i.e., an aircraft must perform one and only one maneuver:

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

$$\sum_{j=1}^{m} x_{ij} = 1 \quad \forall i,\ i = 1 \ldots, n. \tag{2}$$

After proposing the set of maneuvers for each aircraft, we can then fill a 4-dimensional Conflict Matrix (CM) of size $n \times m \times n \times m$ with binary values indicating presence or absence of a conflict between two aircraft,

$$CM(i, j, i', j') = \begin{cases} 1 \text{ if aircraft } i \text{ performing maneuver } j \text{ conflicts} \\ \quad \text{with aircraft } i' \text{ performing maneuver } j' \\ 0 \text{ otherwise.} \end{cases} \tag{3}$$

To detect the potential conflicts, we use a subroutine proposed by Bilimoria [5] wherein we appropriately transform the coordinate system and calculate the relative aircraft speed. Naturally, the entire matrix is redundant due to its symmetry, i.e., $\mathrm{CM}(i, j, i', j') = \mathrm{CM}(i', j', i, j)$.

The primary focus of the tactical deconfliction problem is to modify aircraft trajectories to resolve all conflicts. This objective can be achieved by satisfying the following constraint:

$$\sum_{i=1}^{n} \sum_{j=1}^{m} \sum_{i'=1}^{n} \sum_{j'=1}^{m} x_{ij} x_{i'j'} \mathrm{CM}(i, j, i', j') = 0. \tag{4}$$

We can clearly see that, while it is relatively efficient to check whether the solution is feasible, the number of possible solutions grows exponentially with the number of aircraft and maneuvers.

The aircraft deconfliction problem extends beyond the sole consideration of avoiding conflicts as it also encompasses the optimization of various parameters such as fuel consumption or average delay. Typically, such criteria can be aggregated into a cost function to minimize, comprising partial costs for each aircraft:

$$C = \sum_{i=1}^{n} \sum_{j=1}^{m} C_{ij}. \tag{5}$$

In this work, we simplify the optimization process by focusing solely on minimizing the total number of changes to the original trajectory. Nevertheless, the objective can be easily expanded to incorporate more sophisticated criteria as needed.

2.2 Quantum formulation and encoding

When addressing optimization challenges, quantum computing offers a variety of approaches to choose from [2]. In this study, our emphasis is on two different optimization algorithms, namely the Quantum Approximate Optimization Algorithm (QAOA) [11] and the Quantum Alternating Operator Ansatz

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

(QAOAnsatz) [14]. These two algorithms are rooted in the Adiabatic Theorem [6], which states that a quantum system in an eigenstate undergoing slow enough changes will remain in that eigenstate. The mathematical connection between these algorithms and the Adiabatic Theorem is not rigid. In practice, the process begins with an arbitrary state, preferably an easy-to-prepare ground state [15]. This initial state then evolves into the ground state that corresponds to the solution of the problem described by the problem Hamiltonian. The subsequent discussion outlines how to construct such a Hamiltonian.

For the translation of the formulas derived in Section 2.1 to quantum Hamiltonians we employ the composition rules described in [13]. In this process, we make use of the Pauli matrices: $I = \left(\begin{smallmatrix} 1 & 0 \\ 0 & 1 \end{smallmatrix}\right)$, $Z = \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$. The first constraint, ensuring that an aircraft can perform one and only one maneuver, can be described in the following way:

$$H_1 = \sum_{i=1}^{n} I - \sum_{j=1}^{m} \left(H_x(x_{ij}) \prod_{j'=1, j' \neq j}^{m} (H_{\text{not}}(x_{ij'})) \right). \tag{6}$$

The Hamiltonian term $H_{\text{not}}(x_{ij'}) = \frac{1}{2}(I + Z_{ij'})$ represents a boolean clause that has a value of 1 if aircraft i does not perform maneuver j'. The product represents a clause with a 1 if any other maneuver, except j, is not performed. We specify the clause that has a value of 1 if aircraft i performs maneuver j by the Hamiltonian term $H_x(x_{ij}) = \frac{1}{2}(I - Z_{ij})$. We repeat the process for every possible maneuver j to achieve a boolean clause that has a value of 1 if we have a correct one-hot encoding. Note that we want the ground state to represent the desired solution, so we must negate the Hamiltonian. Afterwards, we sum over all possible aircraft.

The second constraint, ensuring that no two aircraft are in conflict, is represented as follows:

$$H_2 = \sum_{i,j,i',j' : \text{CM}(i,j,i',j')=1} H_{\text{and}}(x_{ij}, x_{i'j'}). \tag{7}$$

The Hamiltonian term $H_{\text{and}}(x_{ij}, x_{i'j'}) = \frac{1}{4}I - \frac{1}{4}(Z_{ij} + Z_{i'j'} - Z_{ij}Z_{i'j'})$ represents a boolean clause that evaluates to 1 only if aircraft i performs maneuver j and aircraft i' performs maneuver j'. Summing these situations gives us the total number of conflicts. Naturally, our objective is to minimize the number of conflicts, aiming for a value of 0.

The optimization criterion is determined by a Hamiltonian that assigns appropriate weights to the chosen maneuvers of each aircraft:

$$H_{\text{opt}} = \sum_{i=1}^{n} \sum_{j=1}^{m} w_{ij} H_x(x_{ij}). \tag{8}$$

Here, w_{ij} represents the cost associated with aircraft i performing maneuver j. When aiming to minimize the number of changes from the original trajectories,

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

the weights for the original trajectories are set to 0, while a positive value is assigned to the weights corresponding to modified trajectories.

These partial Hamiltonians have been crafted to be combined into a final Hamiltonian, where the ground state aligns with our desired deconflicted solution:

$$H = \theta_1 H_1 + \theta_2 H_2 + \theta_{\text{opt}} H_{\text{opt}}. \tag{9}$$

In the final Hamiltonian, we introduced additional multipliers to ensure that the ground state consistently corresponds to a feasible solution, regardless of the number of changes needed in the original trajectory. A simple valid assignment can be made as follows: $\theta_1 = 1,\ \theta_2 = 1,\ \theta_{\text{opt}} = \text{sum(CM)}$, where sum(CM) is the number of all conflicts (all 1s) in the CM.

3 Application

The two algorithms, QAOA and its enhancement, QAOAnsatz, are both hybrid quantum-classical variational algorithms. In these approaches, a parametrized quantum circuit is designed, and the variational parameters are iteratively adjusted using a classical optimizer to minimize the cost function defined by the expectation value of a chosen observable. We provide a brief overview of the foundations of each of these algorithms and their application in solving the tactical deconfliction problem.

3.1 Quantum Approximate Optimization Algorithm

Given R qubits, QAOA initializes by preparing the quantum register in the state $|+\rangle^{\otimes R}$, which is the ground state of a mixing Hamiltonian composed of Pauli-X gates, $H_M = \sum_{i=1}^{R} X_i$. It then alternately applies the problem Hamiltonian (also known as the cost Hamiltonian) and the mixer Hamiltonian to the initial state, p times, where p is a positive integer. The number p is also referred to as the depth of QAOA. The evolution of Hamiltonians is parameterized by two sequences of variational parameters, namely $\overrightarrow{\gamma}$ and $\overrightarrow{\beta}$. The former controls H_c, while the latter controls H_m. Combining these elements, the final state $|\psi\rangle$ after evolution is expressed as follows

$$|\psi_p(\overrightarrow{\gamma}, \overrightarrow{\beta})\rangle = e^{-i\beta_p H_M} e^{-i\gamma_p H_C} \dots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_C} |+\rangle^{\otimes R}. \tag{10}$$

The role of H_c is to distinguish our desired problem solution by applying a change in phase to it. In the context of the tactical deconfliction problem, we simply need to set $H_C = H$, see Equation 9. The H_M, on the other hand, aims to amplify the phase increasing the probability of measuring the desired solution. This is achieved by adjusting the variational parameters using a classical optimizer which minimizes the expectation value:

$$\min_{\overrightarrow{\gamma}, \overrightarrow{\beta}} \langle \psi_p(\overrightarrow{\gamma}, \overrightarrow{\beta}) | H_C | \psi_p(\overrightarrow{\gamma}, \overrightarrow{\beta}) \rangle. \tag{11}$$

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

The expectation value of the circuit measurement is also commonly known as energy. Minimizing the energy is equivalent to increasing the probability of measuring solution to our problem. It is noteworthy that H_c serves a dual purpose, as it also functions as a cost function in this context.

3.2 Quantum Alternating Operator Ansatz

We can modify the approach by initializing the quantum register with a state that corresponds to a feasible solution (or a semi-feasible solution, such as one that satisfies only one of several constraints). The algorithm then applies H_C as usual to distinguish our desired solution in phase, but the mixer Hamiltonian is used differently. It is designed to provide transitions from one feasible solution to another. This way we explore and search for the lowest-energy solution only within a feasible subspace constrained by the hard constraints of our problem, which is the essence of the QAOAnsatz algorithm [14].

In the context of the tactical deconfliction problem, we have chosen to encode only the one-hot constraint (Equation 6) into H_M. To achieve this, we employ a single-qubit ring mixer defined as follows:

$$H_M = \sum_{i=1}^{n} X_{im}X_{i1} + Y_{im}Y_{i1} + \sum_{j=1}^{m} X_{ij}X_{ij+1} + Y_{ij}Y_{ij+1}. \tag{12}$$

Here, the Y symbol represents the Pauli-Y gate. The term $X_{im}X_{i1} + Y_{im}Y_{i1}$ closes the loop between the last and the first qubit, representing the one-hot encoding for each aircraft.

As we have encoded the one-hot constraint into H_M, we can remove the constraint from H_C:

$$H_C = \theta_2 H_2 + \theta_{\text{opt}} H_{\text{opt}}. \tag{13}$$

However, it's important to note that in the presence of noisy hardware, the evolution may drift away from feasible-only solutions. In such cases, having a redundant term in the cost Hamiltonian might be advantageous. In this paper, we choose to use the full cost Hamiltonian, as formulated in Equation 9.

4 Experimental results

In the proposed encoding, the number of qubits was equal to the product of the number of aircraft and their maneuvers. Consequently, instances with an identical number of variables could differ in the ratio of aircraft to maneuvers. We started our set of experiments by investigating how altering these two factors affects instance difficulty. For this purpose, we introduced a set of instances that require only 12 qubits but feature different numbers of aircraft and maneuvers, and these instances were artificially generated by constructing CM to ensure only one solution exists.

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

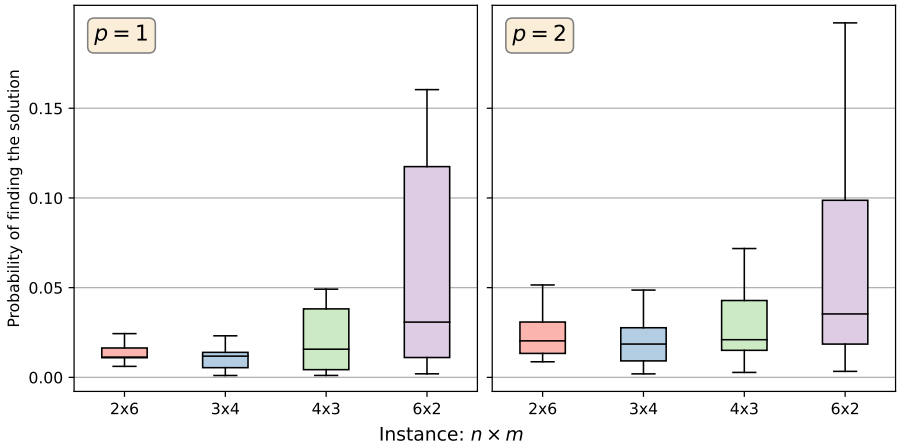


Fig. 2. Average success probability as a function of instance type. Instances are ordered based on the number of aircraft, ranging from 2 aircraft with 6 possible maneuvers to 6 aircraft with only 2 possible maneuvers. The comparison involves two different QAOA depths.

For each instance, we executed 100 QAOA circuits on a noisy simulator with varying initial variational parameters, and the results of success probability were averaged. We used SPSA [24] as the optimizer, as it has proven to perform well on noisy setups. The outcomes are presented in Figure 2. Observing both circuit depths p, we noted that the algorithm faces increasing difficulty in finding the correct solution as the number of maneuvers grows. Conversely, increasing the number of aircraft at the expense of maneuvers tends to make the instance easier. This behavior aligns with our expectations, as ensuring that no two aircraft are in conflict requires less entanglement between qubits compared to constraining that an aircraft can perform one and only one maneuver. More entanglement naturally makes the circuit longer, introducing additional noise. Moreover, entangling gates are typically more error-prone than single-qubit gates. As a side note, we observed that increasing the circuit depth also appears to result in a slight improvement in the average success probability. After conducting initial experiments on a quantum simulator, we evaluated the capabilities of physical quantum computers.

Existing quantum hardware in the noisy intermediate-scale quantum (NISQ) era provides access to several hundred superconducting qubits. Promising qubit implementations use other quantum technologies, such as trapped ions, neutral atoms, or photons. However, the superconducting quantum architectures lack all-to-all qubit connectivity, requiring multiple swaps to make them adjacent before entanglement. Introducing extra SWAP quantum gates may cause additional errors, potentially degrading the solution quality and, in extreme cases, leading to a failure to find one. With this in mind, we decided to downscale

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

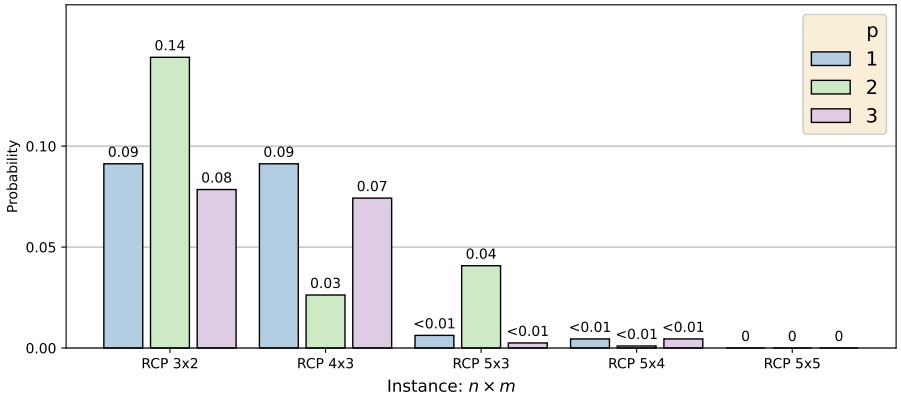


Fig. 3. Probability of finding a solution to the deconfliction problem in the function of instance difficulty and QAOA depth. The instances are the Random Circle Instances with n aircraft, each of the aircraft having m maneuvers to choose from (e.g., $n = 5$, $m = 3$ for RCP 5×3). Experiments were launched on the 133-qubit *ibm_torino*.

the Random Circle Problem (RCP) instances [23] to involve 3, 4 and 5 aircraft. For instance, with 3 aircraft, we proposed 2 maneuvers and for 4 aircraft we proposed 3 maneuvers. The instance with 5 aircraft was approached with 3, 4, and 5 maneuvers. This results in a total of five RCP instances, requiring 9, 12, 15, 20, and 25 qubits, respectively.

We evaluated the performance of the latest IBM quantum computer using the superconduting 133-qubit *ibm_torino* quantum computer in solving all instances across three different QAOA depths. The results are illustrated in Figure 3. Clearly, instances requiring fewer qubits are generally easier to solve. As we move to cases with 5 aircraft, the probabilities of measuring a correct solution drop below 0.01 (less than 1%). It is important to note that this low success probability does not indicate failure, as each circuit is typically measured several thousand times. Given the exponential complexity of the tactical deconfliction problem, achieving a correct solution for even a dozen qubits surpasses the performance of a random guess. Even a single positive outcome is sufficient to solve the considered instance. Unfortunately, the quantum computer selected for our experiments could not solve the problem instance with 5 aircraft and 5 maneuvers. Additionally, we could not identify any noticable trend within the circuit depth, largely due to the inherent randomness of a quantum device. Consequently, further experiments are necessary.

Our final set of experiments involved a comparison between QAOA and QAOAnsatz. Once more, we measured the difference on a quantum simulator and take the average of 100 runs. Given that the tactical deconfliction problem is an optimization problem, we choosen to minimize the number of changes to the original flight schedule. Consequently, we present the probabilities of find-

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

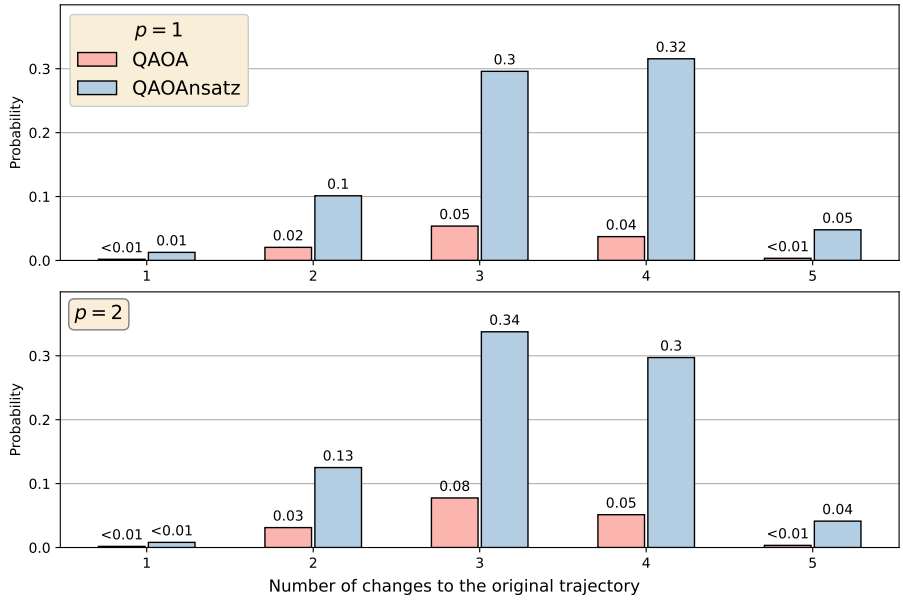


Fig. 4. Comparison between QAOA and QAOAnsatz on a noisy quantum simulator across various two depths, with a focus on the optimization criterion of minimizing changes to the original trajectory. The probabilities of finding a solution to the RCP 5×3 problem are averaged over 100 runs.

ing a correct solution for the RCP 5×3 instance in a function of the number of changes required to achieve a correct solution. The results are shown in Figure 4.

We observed that leveraging the feature of QAOAnsatz, which allows for incorporating hard constraints into the mixer Hamiltonian, provides a significant advantage over using mixers from the vanilla QAOA. The probabilities of measuring a solution to the problem are much higher for QAOAnsatz. However, QAOAnsatz still faces challenges in finding solutions that require only one change to the flight schedule to deconflict aircraft. The experiments demonstrate that QAOAnsatz might be a noteworthy algorithm candidate capable of solving deconfliction instances that QAOA could not handle. We leave this investigation for future work.

5 Conclusions and future work

In this paper, we have successfully shown how to formulate the aircraft deconfliction problem in a way that is applicable to solve using quantum variational algorithms. By designing a proper cost Hamiltonian for the Quantum Approximate Optimization Algorithm (QAOA) and incorporating hard constraints into the mixer Hamiltonian of the Quantum Alternating Operator Ansatz (QAOAnsatz),

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

we have demonstrated the efficacy of quantum computing in addressing this challenge. Our experiments have validated the feasibility of quantum approaches in handling the complexity of aircraft deconfliction and shed light on the nuanced interplay between aircraft and maneuvers in determining solution difficulty. Moreover, using physical quantum machines, such as the IBM quantum computer, has underscored the practicality of our proposed methodologies in real-world settings.

We plan to extend our work in a twofold manner. Firstly, we plan to enhance the series of experiments qualitatively. One intriguing avenue for exploration involves investigating the effects of removing the constraint that limits each aircraft to one and only one maneuver. Suppose an airplane can execute more than one maneuver simultaneously. In that case, it implies that both maneuvers are conflict-free, enabling the decision-making process to be deferred to the post-processing phase. Another way of improving the solution finding would be to perform a more in-depth analysis of QAOAnsatz variants, mainly by incorporating controlled state transitions to the mixer Hamiltonian. We should not neglect the fact to address trajectory recovery, which was considered in some papers.

Secondly, we plan to enhance the series of experiments quantitatively by performing more experiments and trying to solve bigger problem instances. Some of the implemented qualitative measures, e.g. moving the one and only one constraint to the post-processing phase, will naturally allow for performing larger experiments. A notable consequence of the time-dependent three-dimensional domain of the problem is that some maneuvers do not with each other. It means that we can find such a bijection between variables and qubits so that the no-conflicting maneuvers correspond to qubits which are distant from each other on the quantum computer processor topology, which could significantly reduce the need for SWAP gates, suppressing the noise. Finally, performing more experiments on the same size instances would also improve the precision and potential findings of the experimental results.

Acknowledgments. This research was supported by Poznan Supercomputing and Networking Center, project no DOB-SZAFIR/01/B/023/01/2021 and partially funded by Poznan University of Technology, project no 0311/SBAD/0734. The access to quantum devices was possible through IBM Quantum Innovation Center established at Poznan Supercomputing and Networking Center.

References

1. Airports council international europe | aci europe - media_2024 (Feb 2024), https://www.aci-europe.org/media-room/477-passenger-traffic-reaches-nearly-95-of-pre-pandemic-levels-in-2023.html
2. Abbas, A., Ambainis, A., Augustino, B., Bärtschi, A., Buhrman, H., Coffrin, C., Cortiana, G., Dunjko, V., Egger, D.J., Elmegreen, B.G., et al.: Quantum optimization: Potential, challenges, and the path forward. arXiv preprint arXiv:2312.02279 (2023)

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

3. Alonso-Ayuso, A., Escudero, L.F., Martín-Campo, F.J.: Collision avoidance in air traffic management: A mixed-integer linear optimization approach. IEEE Transactions on Intelligent Transportation Systems **12**(1), 47–57 (2010)
4. Alonso-Ayuso, A., Escudero, L.F., Martín-Campo, F.J.: Exact and approximate solving of the aircraft collision resolution problem via turn changes. Transportation Science **50**(1), 263–274 (2016)
5. Bilimoria, K.: A geometric optimization approach to aircraft conflict resolution. In: 18th Applied aerodynamics conference. p. 4265 (2000)
6. Born, M., Fock, V.: Beweis des adiabatensatzes. Zeitschrift für Physik **51**(3-4), 165–180 (1928)
7. Cafieri, S., Durand, N.: Aircraft deconfliction with speed regulation: new models from mixed-integer optimization. Journal of Global Optimization **58**, 613–629 (2014)
8. Cafieri, S., Omheni, R.: Mixed-integer nonlinear programming for aircraft conflict avoidance by sequentially applying velocity and heading angle changes. European journal of operational research **260**(1), 283–290 (2017)
9. Cerulli, M., d'Ambrosio, C., Liberti, L., Pelegrín, M.: Detecting and solving aircraft conflicts using bilevel programming. Journal of Global Optimization **81**, 529–557 (2021)
10. Chai, Y., Epifanovsky, E., Jansen, K., Kaushik, A., Kühn, S.: Simulating the flight gate assignment problem on a trapped ion quantum computer. arXiv preprint arXiv:2309.09686 (2023)
11. Farhi, E., Goldstone, J., Gutmann, S.: A quantum approximate optimization algorithm. arXiv preprint arXiv:1411.4028 (2014)
12. Grönkvist, M.: The tail assignment problem. Citeseer (2005)
13. Hadfield, S.: On the representation of boolean and real functions as hamiltonians for quantum computing. ACM Transactions on Quantum Computing **2**(4), 1–21 (2021)
14. Hadfield, S., Wang, Z., O'gorman, B., Rieffel, E.G., Venturelli, D., Biswas, R.: From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. Algorithms **12**(2), 34 (2019)
15. He, Z., Shaydulin, R., Chakrabarti, S., Herman, D., Li, C., Sun, Y., Pistoia, M.: Alignment between initial state and mixer improves qaoa performance for constrained optimization. npj Quantum Information **9**(1), 121 (2023)
16. Kuchar, J.K., Yang, L.C.: A review of conflict detection and resolution modeling methods. IEEE Transactions on intelligent transportation systems **1**(4), 179–189 (2000)
17. Lehouillier, T., Omer, J., Soumis, F., Desaulniers, G.: Two decomposition algorithms for solving a minimum weight maximum clique model for the air conflict resolution problem. European Journal of Operational Research **256**(3), 696–712 (2017)
18. Martins, L.N., Rocha, A.P., Castro, A.J.: A qubo model to the tail assignment problem. In: ICAART (2). pp. 899–906 (2021)
19. Mohammadbagherpoor, H., Dreher, P., Ibrahim, M., Oh, Y.H., Hall, J., Stone, R.E., Stojkovic, M.: Exploring airline gate-scheduling optimization using quantum computers. arXiv preprint arXiv:2111.09472 (2021)
20. Omer, J.: A space-discretized mixed-integer linear model for air-conflict resolution with speed and heading maneuvers. Computers & Operations Research **58**, 75–86 (2015)

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

21. Pallottino, L., Feron, E.M., Bicchi, A.: Conflict resolution problems for air traffic management systems solved with mixed integer programming. IEEE transactions on intelligent transportation systems **3**(1), 3–11 (2002)
22. Pelegrín, M., d'Ambrosio, C.: Aircraft deconfliction via mathematical programming: Review and insights. Transportation science **56**(1), 118–140 (2022)
23. Rey, D., Hijazi, H.: Complex number formulation and convex relaxations for aircraft conflict resolution. In: 2017 IEEE 56th annual conference on decision and control (cdc). pp. 88–93. IEEE (2017)
24. Spall, J.C.: An overview of the simultaneous perturbation method for efficient optimization. Johns Hopkins apl technical digest **19**(4), 482–492 (1998)
25. Stollenwerk, T., Lobe, E., Jung, M.: Flight gate assignment with a quantum annealer. In: International Workshop on Quantum Technology and Optimization Problems. pp. 99–110. Springer (2019)
26. Stollenwerk, T., O'Gorman, B., Venturelli, D., Mandra, S., Rodionova, O., Ng, H., Sridhar, B., Rieffel, E.G., Biswas, R.: Quantum annealing applied to de-conflicting optimal trajectories for air traffic management. IEEE transactions on intelligent transportation systems **21**(1), 285–297 (2019)
27. Vela, A., Solak, S., Singhose, W., Clarke, J.P.: A mixed integer program for flight-level assignment and speed control for conflict resolution. In: Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference. pp. 5219–5226. IEEE (2009)
28. Vikstål, P., Grönkvist, M., Svensson, M., Andersson, M., Johansson, G., Ferrini, G.: Applying the quantum approximate optimization algorithm to the tail-assignment problem. Physical Review Applied **14**(3), 034009 (2020)

ICCS Camera Ready Version 2024
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-63778-0_22

Publication [P4]

Pecyna, T., Różycki, R.: Improving quantum optimization algorithms by constraint relaxation. Applied Sciences. 14, 8099 (2024). https://doi.org/10.3390/app14188099.

Ministry points / journal: 100

Contribution of authors:

- Tomasz Pecyna:
  - Authorship of the idea underlying the paper.
  - Authorship of the idea of the quantum approach
  - Implementation of the quantum approach
  - Performing experiments, results analysis
  - Writing the text of publication
- Rafał Różycki:
  - Resuts analysis
  - Review and editing of the paper

applied sciences

MDPI

Article

Improving Quantum Optimization Algorithms by Constraint Relaxation

Tomasz Pecyna [1,2] and Rafał Różycki [2,*]

[1] Poznan Supercomputing and Networking Center, IBCH PAS, 61-139 Poznan, Poland
[2] Institute of Computing Science, Poznan University of Technology, 60-965 Poznan, Poland
* Correspondence: rafal.rozycki@cs.put.poznan.pl

Abstract: Quantum optimization is a significant area of quantum computing research with anticipated near-term quantum advantages. Current quantum optimization algorithms, most of which are hybrid variational-Hamiltonian-based algorithms, struggle to present quantum devices due to noise and decoherence. Existing techniques attempt to mitigate these issues through employing different Hamiltonian encodings or Hamiltonian clause pruning, but they often rely on optimistic assumptions rather than a deep analysis of the problem structure. We demonstrate how to formulate the problem Hamiltonian for a quantum approximate optimization algorithm that satisfies all the requirements to correctly describe the considered tactical aircraft deconfliction problem, achieving higher probabilities for finding solutions compared to previous works. Our results indicate that constructing Hamiltonians from an unconventional, quantum-specific perspective with a high degree of entanglement results in a linear instead of exponential number of entanglement gates instead and superior performance compared to standard formulations. Specifically, we achieve a higher probability of finding feasible solutions: finding solutions in nine out of nine instances compared to standard Hamiltonian formulations and quadratic programming formulations known from quantum annealers, which only found solutions in seven out of nine instances. These findings suggest that there is substantial potential for further research in quantum Hamiltonian design and that gate-based approaches may offer superior optimization performance over quantum annealers in the future.

Keywords: quantum computing; quantum optimization; quantum approximate optimization algorithm; tactical aircraft deconfliction problem; quadratic unconstrained binary optimization; Hamiltonian; noisy intermediate-scale quantum era

check for updates

Citation: Pecyna, T.; Różycki, R. Improving Quantum Optimization Algorithms by Constraint Relaxation. *Appl. Sci.* **2024**, *14*, 8099. https://doi.org/10.3390/app14188099

Academic Editor: Sergio Pagano

Received: 30 July 2024
Revised: 27 August 2024
Accepted: 7 September 2024
Published: 10 September 2024

Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

As a relatively new and rapidly evolving field in both science and technology, the full potential of quantum computing remains largely uncharted. Researchers are still exploring its practical usefulness across various domains. While significant advancements have been made, such as Shor's algorithm [1] for factoring integers and Grover's algorithm [2] for unstructured search problems, which demonstrate the theoretical advantages of quantum computing, its real-world applications are still being uncovered.

Among the different branches of quantum computing, one area has demonstrated significant practical potential: quantum annealers. Quantum annealing, a method specifically designed for solving optimization problems, has shown promising results and near-practical utility. Quantum annealers, such as those developed by D-Wave Systems, are being increasingly utilized for their ability to find approximate solutions to complex optimization problems more efficiently than any other quantum architecture paradigm [3]. However, the downside is that quantum annealers are not universal quantum computers. Despite the fact that the vast majority of combinatorial optimization problems can be represented as quadratic unconstrained binary optimization (QUBO) formulations, researchers continue to explore and improve other approaches. With simultaneous advancements in

the quality of universal quantum computers, the true leader in quantum optimization has yet to be determined.

Most quantum optimization algorithms in the NISQ (noisy intermediate-scale quantum) era involve representing the optimization problem as a Hamiltonian [4]. In physics, a Hamiltonian represents the energy of a system and governs its time evolution. In quantum computing, it often serves as a crucial component for determining measurable quantities and finding the most favorable solutions by seeking the system's ground state. Although some general guidelines for constructing such Hamiltonians have been established [5], there is no single, universal method for doing so. The construction of a Hamiltonian for a given problem can vary based on the emphasis placed on different aspects of the problem or by considering different quantum paradigms. For example, quantum annealers are designed to handle optimization problems in QUBO form, which constrains the Hamiltonian to contain entanglements of at most second-degree, i.e., entangling at most two qubits with each other.

Another aspect of encoding the Hamiltonian is to introduce interpretability to the energy values. In its simplest form, the Hamiltonian can be monotonic, meaning that better solutions have lower energy than worse ones. However, the Hamiltonian can also be encoded to provide more detailed information. For instance, the energy value might represent the number of unsatisfied constraints, giving a clear indication of how far a solution is from being feasible. Additionally, each digit or component of the Hamiltonian's energy value could correspond to different constraints or optimization criteria, allowing for a more granular analysis of the solution's quality.

Choosing the encoding for a Hamiltonian should not be considered in isolation from the performance of the algorithm. Hamiltonians that encode more detailed information in their energy values tend to be more complex. This complexity often necessitates the algorithm to invest more time and computational effort to achieve convergence. Additionally, such Hamiltonians may require the construction of more intricate quantum circuits, which involve costly operations and are more susceptible to noise. This is particularly problematic for current quantum computers, which are limited by noise and error rates in the NISQ era. Therefore, a balance must be struck between the richness of the information provided by the Hamiltonian and the practical limitations of the quantum hardware.

In this paper, we focus on a specific subset of gate-based quantum optimization algorithms: namely, variational algorithms. We explore an approach for constructing a Hamiltonian for the standard quantum approximate optimization algorithm (QAOA) [6]. Although we primarily discuss the vanilla QAOA, our approach is also applicable to more recent and advanced versions of QAOA [7], potentially enhancing their performance in the same way as it benefits the standard QAOA.

2. Idea of Better Quantum Optimization

The QAOA is a gate-based hybrid classical quantum algorithm inspired by the adiabatic theorem. The adiabatic theorem states that a quantum system initially in an eigenstate will remain in that eigenstate if the Hamiltonian governing the system changes sufficiently slowly [8]. The core idea of QAOA is to approximate this adiabatic process using a sequence of quantum gates, enabling the transition from an easy-to-prepare initial state to a state that approximates the solution to a given optimization problem. To implement this, QAOA uses a combination of two Hamiltonians: H_C, the problem Hamiltonian, and H_M, the mixing Hamiltonian. The evolution is discretized using the Lie–Trotter product formula [9], resulting in a sequence of alternating unitary operations. The final state of the algorithm after p steps is given by:

$$|\psi_p(\overrightarrow{\gamma}, \overrightarrow{\beta})\rangle = e^{-i\beta_p H_M} e^{-i\gamma_p H_C} \dots e^{-i\beta_1 H_M} e^{-i\gamma_1 H_C} |+\rangle^{\otimes R}. \quad (1)$$

We can clearly see that the problem Hamiltonian serves a dual role in QAOA. Firstly, it acts as a quantum observable, measuring the energy of the system and providing feedback

to adjust the variational parameters $\overrightarrow{\gamma}$ and $\overrightarrow{\beta}$. Secondly, it defines the structure of the quantum circuit, determining the gates and operations required for the algorithm. Because the Hamiltonian directly influences both the optimization process and the quantum circuit's complexity, it is beneficial to simplify and reduce its complexity wherever possible. A simpler Hamiltonian can lead to shorter quantum circuits with fewer entanglement gates, which, in turn, might increase the chances of achieving good results and solving larger problem instances.

The structure of the Hamiltonian is also influenced by the chosen variable encoding of the problem. Often, a binary encoding is employed, where a qubit in the state $|0\rangle$ indicates that a particular option is not selected, and $|1\rangle$ indicates that it is selected. This encoding is straightforward for problems with binary variables. For problems with more than two choices, one-hot encoding is typically used. In one-hot encoding, a string of length m is constructed, where m corresponds to the number of possible choices. Each string is designed to have a Hamming weight of 1, meaning that only one qubit in the string is in the state $|1\rangle$, while all others are in the state $|0\rangle$. This ensures that only one option is chosen out of the m possible choices, providing a clear and unambiguous representation of the selection.

There are two main approaches for encoding the one-hot constraint. The first approach uses the QUBO formulation:

$$H_{\text{QUBO-onehot}}(x) = \left(\sum_{j=1}^{m} H_x(x_j) - 1 \right)^2, \tag{2}$$

where x_j is further replaced using the Pauli-Z term:

$$H_x(x_j) = \frac{1}{2}(I - Z_j), \tag{3}$$

and the Pauli matrices are defined as follows:

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

The encoding from Equation (2) has the advantage of using at most $\frac{n(n-1)}{2}$ entanglements. However, if we need to sum many such partial Hamiltonians as constraints, we cannot derive any meaningful interpretation from the energy value, as this encoding increases quadratically with the Hamming distance from the one-hot encoding.

If we want better interpretability, we can use entanglements with higher degrees, which are unavailable in, e.g., D-Wave quantum annealers, with the following formula:

$$H_{\text{full-onehot}} = I - \sum_{j=1}^{m} \left(H_x(x_j) \prod_{\substack{j'=1 \\ j' \neq j}}^{m} H_{\text{not}}(x_{j'}) \right), \tag{4}$$

where

$$H_{\text{not}}(x_{j'}) = \frac{1}{2}(I + Z_{j'}). \tag{5}$$

Equation (4) evaluates to 1 if and only if the one-hot value is encoded correctly, and it is 0 otherwise. However, a significant disadvantage is that after Hermitian evolution, this approach results in a quantum circuit with many entanglement gates of high degree, introducing substantial noise into the model.

In this paper, we observe that not all encodings known from classical problem definitions are as effective as possible for quantum computing. It might seem counterintuitive, but it turns out that by extending the search space, we actually increase the chances of

measuring a feasible/optimal solution while simultaneously simplifying the Hamiltonian. For example, the one-hot constraint can, in some cases, be replaced by a relaxed "at least one" constraint, which ultimately serves the same purpose but significantly reduces circuit length and noise. We show how to do that using a concrete example in Section 4.

3. Literature Review

Many researchers have investigated reducing noise and improving experimental results through circuit manipulations or truncations. For instance, using imperfect Hamiltonian representations for NISQ-era adiabatic quantum optimization has been shown to yield better results when employing specific techniques [10]. Additionally, several studies have examined the variable encoding procedures, noting that different encodings and embeddings significantly affect circuit performance [11–13].

A particularly interesting branch of research related to our approach involves specific variants of the QAOA algorithm [7]. One notable example is the ADAPT-QAOA algorithm, which iteratively selects operators to enhance the QAOA mixer Hamiltonians based on gradient descent algorithm outputs, thereby reducing the overall number of necessary entanglement gates [14]. Another algorithm, focusing on problem Hamiltonians, draws inspiration from classical neural networks by introducing a quantum dropout approach [15]. This method shows that selectively dropping clauses that define the problem Hamiltonian while maintaining the cost function can enhance QAOA performance.

Our work diverges from these approaches by being the first, to our knowledge, to design a QAOA problem Hamiltonian from the beginning to be simpler than its ideal counterpart. This simplification involves the relaxation of constraints within the problem Hamiltonian, aiming to retain the core structure and characteristics of the original problem while making it more amenable to efficient quantum optimization. By doing so, we strive to balance the complexity of the problem representation with the capabilities of current quantum hardware, which is often limited by noise and decoherence.

4. Exemplary Optimization Problem

In this paper, we consider the tactical aircraft deconfliction problem; however, the following approach can also be used to deal with similar constraints in other problems, such as the job shop scheduling problem [16]. The tactical deconfliction problem involves predicting and resolving conflicts between aircraft in airspace from 5 to 30 min into the future. A conflict is defined as a violation of the safety cylinder of an aircraft by another aircraft. Most approaches to solving this problem involve mixed-integer linear/non-linear programming [17], but there are also some recent quantum approaches.

We base our work on the approach described by Pecyna et al. [18], who first formulated the quantum approach for this problem. For a detailed explanation of the approach, we refer the reader to the original paper. Following this approach, for each of n aircraft, we propose m maneuvers and define a set of $n \times m$ binary variables as follows:

$$X = \{x_{ij} : \ i = 1, \ldots, n, \ j = 1, \ldots, m, \ x_{ij} \in \{0, 1\}\}, \tag{6}$$

where the variable x_{ij} taking the value 1 means that aircraft i is assigned maneuver j; it takes the value 0 otherwise. Each variable directly corresponds to a specific qubit in a quantum environment. From this set of variables, we form two types of constraints. The first constraint ensures there are no conflicts, while the second constraint ensures that an aircraft performs one and only one maneuver. The Hamiltonian representing the no-conflict constraint is originally written as:

$$H_1 = \sum_{i,j,i',j' : \mathrm{CM}(i,j,i',j')=1} H_{\mathrm{and}}(x_{ij}, x_{i'j'}). \tag{7}$$

where the CM matrix is defined as in [18]: $H_{\mathrm{and}}(x_{ij}, x_{i'j'}) = \frac{1}{4} I - \frac{1}{4}(Z_{ij} + Z_{i'j'} - Z_{ij} Z_{i'j'})$.

The Hamiltonian for the constraint that ensures an aircraft can perform one and only one maneuver is originally formulated as:

$$H_2 = \sum_{i=1}^{n} I - \sum_{j=1}^{m} \left(H_x(x_{ij}) \prod_{\substack{j'=1 \\ j' \neq j}}^{m} H_{\text{not}}(x_{ij'}) \right), \tag{8}$$

where $H_{\text{not}}(x_{ij'}) = \frac{1}{2}(I + Z_{ij'})$ and $H_x(x_{ij}) = \frac{1}{2}(I - Z_{ij})$.

If these two types of constraints are satisfied, it means that aircraft are assigned conflict-free trajectories and each aircraft performs only one maneuver. The deconfliction problem that is formulated this way does not have an optimization function, so a solution satisfying these two types of constraints is a correct, feasible solution to the problem.

In this work, we focus on improving the second constraint. Note that due to the products in Equation (8), the number of summands containing Pauli-Z terms is $n2^m$, with most necessitating entanglements of degree higher than two. This, in turn, can make it challenging to find the ground state of the final Hamiltonian and can make the quantum circuit prone to errors in noisy quantum environments.

Classical optimization of hard problems almost always benefits from reducing the search space. Setting up more constraints while simultaneously keeping the number of feasible solutions constant prevents classical algorithms from wasting their computing cycles on exploring unprofitable dead ends that do not yield valuable solutions. In quantum computing, however, qubits can be put into superposition, allowing operations on many states simultaneously. This enables virtually cost-free computation on unfeasible solutions without adding any additional overhead. Considering this, we note that the one-hot constraint for the tactical aircraft deconfliction problem is superfluous when using quantum optimization algorithms, and removing it can lead to significant performance improvements.

Counterintuitively, let us replace the constraint that an aircraft can perform one and only one maneuver with the constraint that an aircraft must perform at least one maneuver. The no-conflict constraint remains intact. Feasible solutions would then include solutions with aircraft performing multiple maneuvers simultaneously, which is obviously impossible in real-life situations. This, however, does not concern us, because we can interpret the solution of an aircraft performing multiple maneuvers as if all those maneuvers were conflict-free. The solution would then be a subset of possible non-conflict maneuvers for the aircraft, with the specific maneuver selection postponed until the post-processing phase. Moreover, at this stage, one can define an optimization criterion and select such a solution that optimizes the criterion.

As promising as it might sound, this approach does not help us much. The reason is that if we encoded a Hamiltonian for the at least one maneuver constraint, the number of Pauli-Z summands would be the same as for the one-hot constraint.

We can shift our perspective and identify the fundamental criteria that our constraints must meet:

1. The state $0 \ldots 0$ must have a value of 1;
2. Each state with a Hamming weight of 1, representing the feasible solution, must yield a value of 0;
3. All other possible states must have a non-negative value to ensure that no bitstrings have a lower Hamiltonian value than any correct solution.

These three requirements do not correspond to the one-hot or the at-least-one constraint. However, they are sufficient to form a sensible constraint that contains all essential and indelible requirements for a correct Hamiltonian for the tactical deconfliction problem.

There is one function that directly meets these criteria, which is the multi-variable NOT XOR function. For example, consider an aircraft with $m = 5$ possible maneuvers. The Hamming weights for all $2^6 = 32$ bitstrings would range from 0 to 5. Moreover,

- Bitstrings with Hamming weights of 0, 2, and 4 would yield a value of 1, fulfilling the first requirement and partially the third requirement;
- Bitstrings with Hamming weights of 1, 3, and 5 would yield a value of 0, fulfilling the second requirement and completing the third requirement.

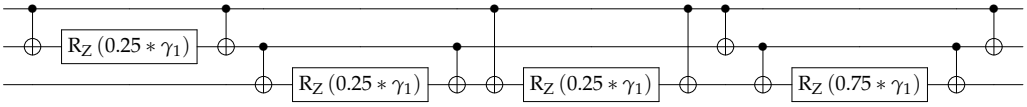
We present a more visual representation of the behavior of the NOT XOR function in Table 1. Subsequently, the Hamiltonian for the NOT XOR function can be encoded as follows:

$$H_{\text{NOT XOR}} = -\tfrac{1}{2}I + \tfrac{1}{2}Z_1Z_2 \ldots Z_m. \tag{9}$$

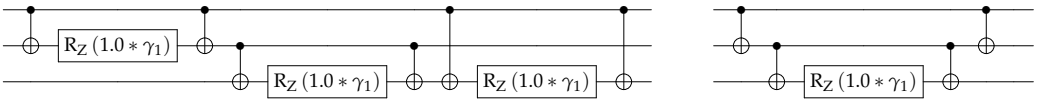
Table 1. Example of possible maneuver assignments for one aircraft and the corresponding evaluation of the NOT XOR function.

Number of Maneuvers (Hamming Weight)	**0**	**1**	**2**	**3**	**4**	**5**
Bitstring	00000	00001	00011 00101	00111 01011	01111	11111
		00010	00110 01001	01101 01110	10111	
		00100	01010 01100	10011 10101	11011	
		01000	10001 10010	10110 11001	11101	
		10000	10100 11000	11010 11100	11110	
NOT XOR	1	0	1	0	1	0

Using the NOT XOR function, we see that there is only one Pauli-Z term with an entanglement of degree m. This represents a significant improvement, as the number of Pauli-Z terms directly corresponds to the number of entanglement gates needed to encode such Hamiltonians in a quantum circuit, as shown in Figure 1. For an aircraft with m possible maneuvers, the one-hot encoding requires $\mathcal{O}(2^m)$ entanglements, the quadratic encoding requires $\mathcal{O}(n^2)$ entanglements, while the NOT XOR encoding requires only $\mathcal{O}(1)$ entanglements.



(**a**) One-hot (standard) encoding



(**b**) Quadratic (QUBO) encoding (**c**) NOT XOR encoding

Figure 1. Quantum circuit representations resulting from Hamiltonian evolution for three different encodings of an aircraft (n = 1) with m = 3 possible maneuvers. The symbol R_Z represents a parameterized quantum gate rotation around the Z-axis.

5. Computational Experiment

Having introduced the NOT XOR Hamiltonian, we compare it with standard one-hot encoding with many entanglements of high degree and with quadratic encoding, which uses only second-degree entanglements. We benchmark these three approaches against nine artificially generated instances of the same size: $n = 3$ aircraft with $m = 5$ alternative maneuvers. The difference between these instances lies in the number of potential conflicts

between maneuvers, resulting in a varying number of feasible solutions to the problem. We choose to test instances with difficulties of 1, 3, 5, 10, 20, 30, 50, 100, and 124 solutions. The reason for choosing such a set of instances is that the first constraint for conflict avoidance (see Equation (7)) already contains the quadratic Pauli-Z terms. Thus, with an increasing number of potential conflicts, the entanglement complexity of the constraint determining the number of maneuvers performed by the aircraft may lose its significance.

The test instances are generated using an iterative graph-based algorithm that detects cycles between nodes. Each node in the graph represents an aircraft performing a maneuver, while edges represent possible coexistence between these maneuvers. By removing an edge between two nodes, we introduce a conflict, reducing the number of solutions. Starting from a complete graph with no conflicts, the number of solutions removed corresponds to the number of cycles that include removing an edge.

We tested the instances in a noisy simulator environment using the noise model derived from the *ibm_torino* quantum device. The 3×5 size of instances was the largest we could compute within a reasonable time frame and required several days of HPC computations across multiple nodes. For the optimization of QAOA parameters, we used the Constrained Optimization BY Linear Approximation (COBYLA) [19] algorithm with the default 1000 iteration threshold. We fixed 10 random initial points, optimized each instance starting from each of these random points, and averaged the results. The results are presented in Figure 2.

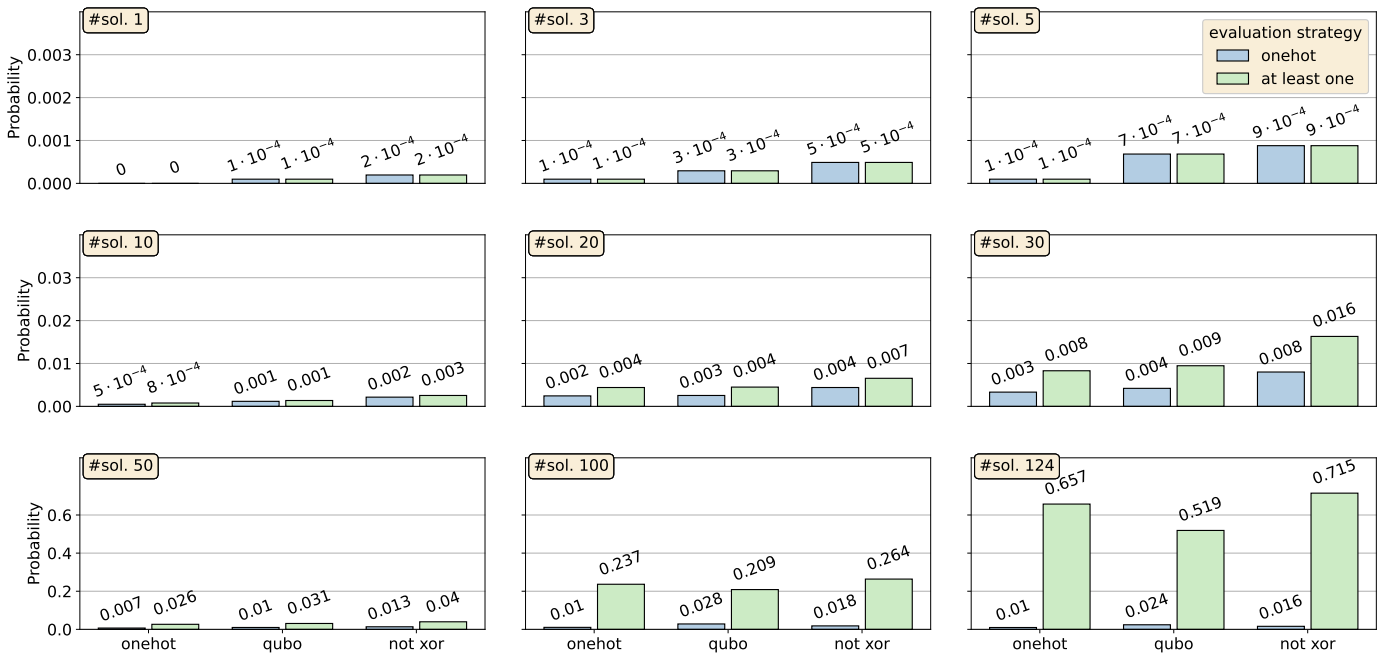


Figure 2. Probabilities of measuring a feasible solution as a function of the number (denoted by #) of feasible solutions in the instance and the chosen encoding. The probabilities are evaluated in two ways. First, we sum the probabilities for all correct one-hot-measured bitstrings (blue bars). Second, we sum the probabilities for all bitstrings that correspond to the aircraft performing at least one maneuver (green bars). After performing a set of measurements, each probability bar represents the ratio of feasible solutions, as defined by the aforementioned rules, to the total number of measurements.

The figure answers two questions: Firstly, is the NOT XOR encoding better in terms of the number of feasible solutions we can find, i.e., does it make the optimization landscape simpler? Secondly, does the NOT XOR encoding introduce less noise to the circuit? One could argue that it would be unfair and biased to report that the NOT XOR solution gives better results when we treat all solutions that have no conflicts and where aircraft perform at least one maneuver as feasible. This is because optimizers with the one-hot and QUBO

Hamiltonians as observables do not aim to increase the probability of measuring states where aircraft perform more than one maneuver. For this purpose, we also evaluate all these Hamiltonians against the one-hot evaluation strategy, which is theoretically not favorable from the perspective of an optimizer aiming to minimize the energy measured with the Hamiltonian that encodes the NOT XOR observable.

We report that in the considered instances, the NOT XOR Hamiltonian outperforms the standard one-hot Hamiltonian in both evaluation strategies. This means that using this kind of encoding is always favorable, as it both simplifies the optimization landscape and reduces potential noise. This also indicates that this approach would be preferred over the standard one-hot encoding even if the goal was to fully satisfy the one-hot encoding, i.e., it was not possible to choose one maneuver for an aircraft out of many at the postprocessing stage.

Moreover, for all instances, the NOT XOR Hamiltonian achieves better results than the QUBO Hamiltonian for the evaluation strategy where bitstrings of aircraft performing at least one maneuver are considered feasible. Additionally, for seven out of nine instances, the NOT XOR Hamiltonian outperforms the QUBO Hamiltonian when we consider bitstrings satisfying the one-hot encoding as feasible. The two instances where the QUBO encoding outperforms the NOT XOR encoding are the two easiest instances, i.e., the two instances with 100 and 124 solutions, which also means fewer conflicts. This result can be easily explained by noting that when there are few possible feasible solutions in an instance, most of them have a Hamming weight of 1. Conversely, when there are many feasible solutions, there are also many solutions having a Hamming weight greater than 1.

6. Conclusions and Future Work

In this paper, we have presented an alternative approach to formulating the problem Hamiltonian for the quantum approximate optimization algorithm that reduces the total number of necessary entanglements from exponential to linear. This formulation employs high-degree entanglements (higher than quadratic), resulting in significant performance benefits, such as fewer noise-induced errors and a higher probability of measuring the correct solution. Our solution outperforms the standard Hamiltonian formulation for the tactical deconfliction problem in all considered instances and surpasses the quadratic formulation known from quantum annealers in seven out of nine instances.

We see two evident directions for further research. Firstly, our experiments were conducted on a noisy simulator designed to closely resemble real quantum machines. However, there are quantum architectures, such as ion traps, that are particularly well-suited for performing high-degree entanglements in a single operation. It would be beneficial to repeat these experiments on real hardware that is optimized for such setups. Secondly, we have demonstrated only one use case of this Hamiltonian formulation. It would be valuable for the community to conduct in-depth research on various well-known optimization problems and explore how to reformulate existing encodings to enhance their efficiency. A key area of interest would be to propose a set of general rules for constructing custom Hamiltonians.

Author Contributions: Conceptualization, T.P.; methodology, T.P. and R.R.; validation, T.P.; formal analysis, T.P. and R.R.; investigation, T.P.; writing original draft, T.P.; editing, T.P. and R.R.; visualization, T.P.; supervision, R.R.; project administration, R.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was financed under the Program of the Polish Ministry of Science and Higher Education "Applied Doctorate" realized in years 2020–2024 (agreement No. DWD/4/24/2022 dated on 6.11.2020) and by the Poznan University of Technology (project No. 0311/SBAD/0746).

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Shor, P.W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* **1999**, *41*, 303–332. [CrossRef]
2. Grover, L.K. A fast quantum mechanical algorithm for database search. In Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, Philadelphia, PA, USA, 22–24 May 1996; pp. 212–219.
3. King, A.D.; Nocera, A.; Rams, M.M.; Dziarmaga, J.; Wiersema, R.; Bernoudy, W.; Raymond, J.; Kaushal, N.; Heinsdorf, N.; Harris, R.; et al. Computational supremacy in quantum simulation. *arXiv* **2024**, arXiv:2403.00910.
4. Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2018**, *2*, 79. [CrossRef]
5. Hadfield, S. On the representation of Boolean and real functions as Hamiltonians for quantum computing. *ACM Trans. Quantum Comput.* **2021**, *2*, 1–21. [CrossRef]
6. Farhi, E.; Goldstone, J.; Gutmann, S. A quantum approximate optimization algorithm. *arXiv* **2014**, arXiv:1411.4028.
7. Blekos, K.; Brand, D.; Ceschini, A.; Chou, C.H.; Li, R.H.; Pandya, K.; Summer, A. A review on quantum approximate optimization algorithm and its variants. *Phys. Rep.* **2024**, *1068*, 1–66. [CrossRef]
8. Born, M.; Fock, V. Beweis des adiabatensatzes. *Z. Für Phys.* **1928**, *51*, 165–180. [CrossRef]
9. Trotter, H.F. On the product of semi-groups of operators. *Proc. Am. Math. Soc. USA* **1959**, *10*, 545–551. [CrossRef]
10. Young, K.C.; Blume-Kohout, R.; Lidar, D.A. Adiabatic quantum optimization with the wrong Hamiltonian. *Phys. Rev. A—At. Mol. Opt. Phys.* **2013**, *88*, 062314. [CrossRef]
11. Di Matteo, O.; McCoy, A.; Gysbers, P.; Miyagi, T.; Woloshyn, R.; Navrátil, P. Improving Hamiltonian encodings with the Gray code. *Phys. Rev. A* **2021**, *103*, 042405. [CrossRef]
12. Leng, J.; Li, J.; Peng, Y.; Wu, X. Expanding Hardware-Efficiently Manipulable Hilbert Space via Hamiltonian Embedding. *arXiv* **2024**, arXiv:2401.08550.
13. Gogeißl, M.; Safi, H.; Mauerer, W. Quantum Data Encoding Patterns and their Consequences. In Proceedings of the 1st Workshop on Quantum Computing and Quantum-Inspired Technology for Data-Intensive Systems and Applications, Santiago, Chile, 9–15 June 2024; pp. 27–37.
14. Zhu, L.; Tang, H.L.; Barron, G.S.; Calderon-Vargas, F.; Mayhall, N.J.; Barnes, E.; Economou, S.E. Adaptive quantum approximate optimization algorithm for solving combinatorial problems on a quantum computer. *Phys. Rev. Res.* **2022**, *4*, 033029. [CrossRef]
15. Wang, Z.D.; Zheng, P.L.; Wu, B.; Zhang, Y. Quantum dropout for efficient quantum approximate optimization algorithm on combinatorial optimization problems. *arXiv* **2022**, arXiv:2203.10101.
16. Kurowski, K.; Pecyna, T.; Slysz, M.; Różycki, R.; Waligóra, G.; Węglarz, J. Application of quantum approximate optimization algorithm to job shop scheduling problem. *Eur. J. Oper. Res.* **2023**, *310*, 518–528. [CrossRef]
17. Pelegrín, M.; d'Ambrosio, C. Aircraft deconfliction via mathematical programming: Review and insights. *Transp. Sci.* **2022**, *56*, 118–140. [CrossRef]
18. Pecyna, T.; Kurowski, K.; Rózycki, R.; Waligóra, G.; Węglarz, J. Quantum Variational Algorithms for the Aircraft Deconfliction Problem. In *Proceedings of the International Conference on Computational Science*; Springer: Berlin/Heidelberg, Germany, 2024; pp. 307–320.
19. Powell, M.J. A direct search optimization method that models the objective and constraint functions by linear interpolation. In *Advances in Optimization and Numerical Analysis*; Springer: Berlin/Heidelberg, Germany, 1994; pp. 51–67.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.

Publication [P5]

Pecyna, T., Siera, D., Bosak, B.: QCG-QuantumLauncher: a modular tool for quantum scenarios. Conference materials from 15th International Conference on Parallel Processing & Applied Mathematics, Ostrava, Czech Republic, September 8-11, 2024

Ministry points / conference: 20

Contribution of authors:

- Tomasz Pecyna:
  - Authorship of the idea underlying the paper.
  - Co-implementation of the software
  - Co-authorship of the text of publication (Sections: Introduction, QCG-QuantumLauncher: future directions, Conclusions)
- Dawid Siera:
  - Co-implementation of the software
  - Co-authorship of the text of publication (Sections: QCG-QuantumLauncher)
- Bartosz Bosak:
  - Consultation on the implementation of the software
  - Co-authorship of the text of publication (Sections: QCG-QuantumLauncher: future directions)

QCG-QuantumLauncher: a modular tool for quantum scenarios

Tomasz Pecyna[0000−0001−7473−3360], Dawid Siera[0009−0005−1626−9658], and Bartosz Bosak[0000−0002−9331−3166]

Poznań Supercomputing and Networking Center, IBCH PAS
{tpecyna,dsiera,bbosak}@man.poznan.pl

Abstract. The evolving landscape of quantum computing at its early stages presents challenges in predicting which from among many different quantum architectures will become dominant. This uncertainty and diversity has led to the proliferation of various software solutions, resulting in complexity for developers and researchers experimenting with different quantum paradigms. Existing tools attempt to address these challenges, yet they often fall short of meeting the needs of current researchers seeking simplicity in software while retaining access to specific quantum hardware details. In this paper, we introduce QCG-QuantumLauncher, a software library designed to solve specific problems by launching selected quantum algorithms on chosen quantum devices through a simple, uniform interface. Additionally, we delineate the future development trajectories of QCG-QuantumLauncher, aiming to position it as the premier tool choice for any quantum researcher.

Keywords: Quantum Computers · Quantum Software · Optimization

1 Introduction

The rapid advancement of quantum technologies, spanning both software and hardware, is undeniable. Motivated by early theoretical breakthroughs demonstrating quantum computational advantages, such as Shor's [22] and Grover's [10] algorithms, enabled by phenomena like superposition, entanglement, and tunneling [20], companies are dedicating substantial time and resources to establish their foothold in this burgeoning field. Their work is further accelerated by the growing challenges faced by current classical architectures, including issues like high energy consumption, heat dissipation, miniaturization constraints, and the inability to maintain Moore's Law, which results in failure to address current computational challenges [6]. Notable players in this arena include Xanadu with their photonic quantum devices [2], D-Wave pioneering in quantum annealers [14], IonQ advancing ion-trap quantum computers [3], and IBM leveraging superconducting technology for their quantum computers [18]. The most promising quantum technology is yet to be determined, leading to a diverse array of approaches. One notable consequence of this diversity is the need for each technology to have a device-specific set of control instructions, often accompanied by

the development of high-level programming libraries to facilitate device programming. While researchers using solely gate-based quantum computers encounter relatively few issues switching between algorithms and quantum computer vendors, thanks to common open standards for circuit definition [4], the same cannot be said for those engaging in cross-architecture research using also other quantum architectures like annealers or bosonic samplers. Such research efforts often encounter unnecessary code redundancy, especially during the preprocessing, launching, data gathering, and postprocessing phases.

Various approaches aim to address this issue. One strategy involves constructing a high-level platform that provides users with specific tools for constructing and managing quantum solutions. This approach aims to shield users from the complexities of different environments, exemplified by platforms like QuantumPath [12]. However, this approach may limit users' control over dataflow and the execution of complex scenarios or custom workflows. On the other hand, tools like XACC [17], QCOR [19] or Quingo [8] offer high control over low-level instructions while maintaining hardware agnosticism. However, these tools lack higher-level modularity, which is essential for less advanced users to easily test their applications on different algorithms and backends with minimal coding. Occupying a middle ground between these two approaches are solutions like Covalent [5] or Cuda-Q [23], offering user-friendly interfaces for managing both classical and quantum resources while still providing sufficient control to develop custom workflows. However, at present, they do not support heterogeneous quantum workflows.

In this paper, we introduce QCG-QuantumLauncher (QCG-QL), a quantum enhancement of the QCG middleware[1] [21] developed by Poznan Supercomputing and Networking Center. QCG-QL is a library for heterogeneous quantum-classical computing, designed to meet the needs of users seeking to solve specific problems by launching selected quantum algorithms on chosen quantum devices. Primary design goals of QCG-QL include the following:

- to cater to quantum software engineers with domain knowledge, ensuring the library is highly useful for their needs;
- to encompass not only gate-based quantum technologies but also various architectures such as quantum annealing or boson sampling;
- to eliminate the need for code redundancy when preparing experiments with the same problems but different algorithms and computers;
- to ensure code reusability and extensibility;
- to maintain consistent, clear, and simple interfaces, enabling less advanced users to fully benefit from using the library.

The details of our quantum software and future extensions will be discussed in the remainder of this paper.

[1] `https://qcg.psnc.pl/`

2 QCG-QuantumLauncher

In the realm of quantum computing optimization, certain patterns consistently emerge, underscoring the necessity of a structured and methodical approach. Initially, it is necessary to explicitly define the problem to be solved. This involves clearly outlining the task at hand and ensuring that the objectives are unambiguous. A well-defined problem statement lays the foundation for subsequent steps by directing the focus towards the specific challenge that needs to be addressed.

Once the problem is clearly defined, the next fundamental step is to identify an algorithm that is optimally suited to solve this specific problem. The selection of an algorithm requires a deep understanding of both the problem characteristics and the available quantum algorithms. In some cases, it might be necessary to design a new algorithm tailored to the unique requirements of the problem. This phase is important as the efficiency and effectiveness of the solution heavily depend on the appropriateness of the chosen algorithm.

Finally, the execution of the algorithm necessitates appropriate hardware. Quantum algorithms require quantum computers or simulators capable of handling the computations. This includes not only the quantum processing units but also the associated classical infrastructure to manage and support quantum operations. The choice of hardware can significantly increase the probability of measuring desired solution and impact the overall performance.

In summary, the path to effective quantum computing involves a well-defined problem statement, the selection of a suitable algorithm, and the execution on appropriate hardware. These three components are interdependent and important for the optimization process.

GCG-QuantumLauncher framework is designed to simplify the complexities of quantum programming by organizing it into these three fundamental components: the problem to be solved, the algorithm used and the hardware utilized to execute the solution. The high-level structure of our approach is shown in Figure 1. This division aims to clarify each component's responsibility. With this unified approach, our program becomes not only easier to write but also easier to read and maintain.

2.1 Architecture

Problem: When solving a problem using a computer, it is essential to define the problem in a manner that the computer can process. This entails specifying both the nature of the problem and the approach for solving it. The same methodology applies to quantum computers. Typically, problems are defined using models such as Quadratic Unconstrained Binary Optimization, Ising models, or Hamiltonians [15]. These models are integral to a wide range of algorithms and can be translated between each other in polynomial time relative to the number of variables involved. In the case of QCG-QL, users need only to prepare a formulation in one of these formats, and the framework will handle the translation to other models.

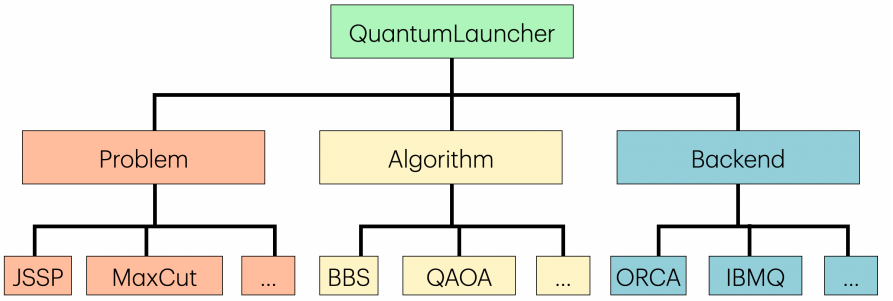


Fig. 1. Diagram illustrating the hierarchical structure of the QCG-Quantum Launcher methodology, highlighting the specific components within each category.

Algorithm: There are many different quantum approaches for solving computational problems on quantum computers. These methods range from the most popular ones, such as Quantum Approximate Optimization Algorithm (QAOA) [7] and Quantum Annealing [13], to their variants like FALQON [16] and Quantum Alternating Operator Ansatz [11]. While each of these algorithms is implemented differently, they share many common similarities. Proper structure can greatly enhance these methods. For example, in QAOA, even the smallest changes can significantly affect the algorithm's performance. A well-structured approach provided by QCG-QL makes it easier to implement and test these small modifications, resulting in more modular and maintainable code.

Backend: Each of the abovementioned algorithms needs to be executed on some computational platform. These platforms can vary widely, offering diverse APIs. To simplify the user experience, this module aims to standardize access to these platforms, regardless of their specific characteristics. The backend module simplifies the process of switching between different platforms, allowing users to easily test their applications on various devices. Many hardware providers allow users to select the machine by specifying its name as a parameter, while others require different functions, which can be complex. Our solution consolidates this process into a single, unified interface, making it more straightforward for the user.

QuantumLauncher: QuantumLauncher module integrates all other modules within the framework, providing a unified interface for users. This integration enhances user convenience by abstracting complex operations and presenting a cohesive workflow from problem definition to solution analysis. By managing these processes centrally, the Quantum Launcher module simplifies the user experience, making quantum programming more accessible and efficient. Additionally, it offers tools for postprocessing and data management, further simplifying the overall workflow.

2.2 Usage

The QCG-QuantumLauncher framework is designed to cater to a broad spectrum of use cases, from educational purposes to advanced research. Its simplicity and intuitive syntax are particularly beneficial for those who are new to quantum computing. Listings 1.1 and 1.2 demonstrate sample code snippets that highlight the ease of use for inexperienced users, providing them with a gentle introduction to quantum programming concepts.

```
import quantum_launcher as ql
from quantum_launcher.qiskit_routines import
      FALQON, QiskitBackend


pr = ql.problems.JSSP(instance_name='toy',
      max_time=3)
alg = FALQON(delta_t=0.03, beta_0=0, n=2)
backend = QiskitBackend('ibm_sherbrooke')
launcher = ql.QuantumLauncher(pr, alg, backend)
launcher.process(save_pickle=True)
```

Listing 1.1. Solving the Job Shop Scheduling Problem with the QAOA Algorithm using *ibm_sherbrooke* Quantum Computer.

```
import quantum_launcher as ql
from quantum_launcher.orca_routines import
      BinaryBosonic, OrcaBackend


pr = ql.problems.QATM(onehot='exact',
      instance_name='RCP_5')
alg = BinaryBosonic()
backend = OrcaBackend('local')
launcher = ql.QuantumLauncher(pr, alg, backend)
launcher.process(save_json=True)
```

Listing 1.2. Solving the QATM Problem with the Binary Bosonic Sampling [9]. Algorithm on a local simulator of the ORCA Quantum Device.

Inexperienced users For individuals beginning their journey in quantum computing, the challenge often lies in integrating various components of the system independently. They must choose a quantum platform, learn the syntax of its specific library, familiarize themselves with the available algorithms, and understand how to formulate a problem in a manner that a quantum computer can process.

The Quantum Launcher framework simplifies this learning process. Users can focus on the aspects they wish to learn first and obtain meaningful results regardless of their starting point or prior knowledge of implementation details. This approach allows users to progressively learn quantum programming without the immediate need to implement complex functions that may be initially beyond their understanding.

Advanced researchers Experienced users can also benefit significantly from the QCG-QuantumLauncher framework. The simplified process of testing and benchmarking new solutions allows researchers to quickly deploy and evaluate their algorithms on multiple machines at the same time, without the need to learn the specifics of each architecture or prepare distinct code for each platform. This not only reduces the amount of code that needs to be written but also enhances the reliability of benchmarks by facilitating direct comparisons across different solutions. The QCG-QL framework is designed as a modular structure, which makes it highly extensible and simplifies the implementation of new features. Additionally, researchers can reuse existing code to develop new, slightly modified solutions. This capability can lead to substantial improvements in the quality of results with minimal effort, enabling efficient postprocessing analysis of the solutions.

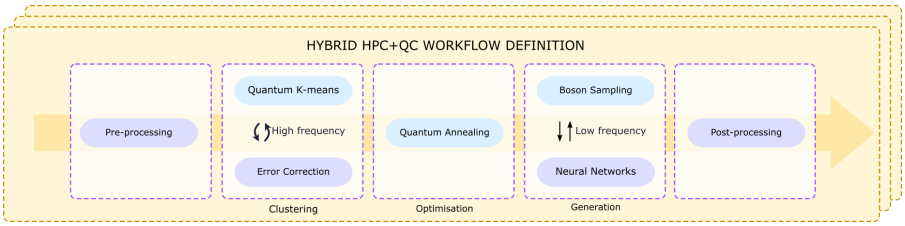


Fig. 2. A sample extended workflow that will be possible to define in a future version of QCG-QL. The workflow should be read from left to right, with the vertical dimension representing parallel hybrid execution. It is worth noting that this workflow utilizes various quantum architectures and establishes two different couplings between classical and quantum resources.

3 QCG-QuantumLauncher: future directions

QCG-QL is currently fully functional and ready for use. However, ongoing development aims to expand its capabilities. At present, QCG-QL supports a single type of workflow: solving a problem with an algorithm on a selected quantum backend. To meet the diverse needs of researchers, our primary focus for enhancement is to make QCG-QL a universally adaptable tool for defining various hybrid workflows. An example of an extended workflow that includes multiple steps can be seen in Figure 2. For this example workflow to be implemented, two necessary developments are required. Firstly, a method for defining specific couplings must be developed. While low-frequency loose coupling is often straightforward, strong coupling e.g. for error correction requires additional software support for high-frequency communication. Secondly, we need to design and develop a common, universal data structure that is agnostic to both quantum architecture and algorithm. Additionally, a standardized method for passing data between consecutive steps of the workflow, involving classical and quantum processing, needs to be established.

Extended multi-step workflows are not the only improvements planned for QCG-QL. Some hybrid workflows are based on algorithms that change their resource requirements during execution. To address this, we intend to integrate QCG-QL with existing tools, such as QCG-PilotJob [1], that will allow us to define more sophisticated and highly dynamic workflows. Furthermore, our team is actively working on integrating QCG-QL with popular frameworks such as IBM's Qiskit, Cuda-Q, and MPI, aiming to make it a versatile choice for seamless connections with external tools. Additionally, we plan to explore possible integrations with graphical job management, monitoring, and data management tools, such as those provided by QCG, to further enhance QCG-QL's capabilities.

There is one more area of particular interest to us, which is especially relevant in data centres and high-performance computing environments. The connection between Artificial Intelligence (AI) and quantum computing has already demonstrated its potential and is believed to create synergy that can result in powerful

models. Therefore, achieving comprehensive functionality in QCG-QL necessitates the incorporation of support for multi-QPU and multi-GPU execution. Furthermore, the integration can be enhanced with asynchronous execution, allowing for greater flexibility and efficiency in executing various quantum AI workflows.

4 Conclusions

In this paper, we provided an overview of the current landscape of software supporting heterogeneous quantum environments. We highlighted the lack of appropriate software and emphasized the need for a comprehensive solution to address this gap. We introduced QCG-QuantumLauncher, a versatile library enabling users to select and manage various quantum problems, devices, and algorithms through a unified interface. Additionally, we discussed the necessary enhancements required to make QCG-QL universally applicable for all use cases. Looking forward, our roadmap for further development includes specific plans to enhance QCG-QL's functionality and usability, ensuring its effectiveness in advancing quantum computing research.

Acknowledgments. Coś tutaj piszemy?

References

1. Bosak, B., Piontek, T., Karlshoefer, P., Raffin, E., Lakhlili, J., Kopta, P.: Verification, validation and uncertainty quantification of large-scale applications with qcg-pilotjob. In: International Conference on Computational Science. pp. 495–501. Springer (2021)
2. Bourassa, J.E., Alexander, R.N., Vasmer, M., Patil, A., Tzitrin, I., Matsuura, T., Su, D., Baragiola, B.Q., Guha, S., Dauphinais, G., Sabapathy, K.K., Menicucci, N.C., Dhand, I.: Blueprint for a scalable photonic fault-tolerant quantum computer. Quantum **5**, 392 (Feb 2021). https://doi.org/10.22331/q-2021-02-04-392, http://dx.doi.org/10.22331/q-2021-02-04-392
3. Chen, J.S., Nielsen, E., Ebert, M., Inlek, V., Wright, K., Chaplin, V., Maksymov, A., Páez, E., Poudel, A., Maunz, P., et al.: Benchmarking a trapped-ion quantum computer with 29 algorithmic qubits. arXiv preprint arXiv:2308.05071 (2023)
4. Cross, A., Javadi-Abhari, A., Alexander, T., De Beaudrap, N., Bishop, L.S., Heidel, S., Ryan, C.A., Sivarajah, P., Smolin, J., Gambetta, J.M., et al.: Openqasm 3: A broader and deeper quantum assembly language. ACM Transactions on Quantum Computing **3**(3), 1–50 (2022)
5. Cunningham, W., Esquivel, A., Jao, C., Hasan, F., Bala, V., Sanand, S., Venkatesh, P., Tandon, M., Ochia, O.E., Rosen, A.S., dwelsch esi, jkanem, Aravind, HaimHorowitzAgnostiq, Li, R., Neagle, S.W., valkostadinov, Ghukasyan, A., Rao, P.U., Dutta, S., WingCode, Hughes, A., RaviPsiog, Udayan, Akalanka, Obasi, A., Singh, D., FilipBolt: Agnostiqhq/covalent: v0.228.0-rc.0 (Sep 2023). https://doi.org/10.5281/zenodo.8369670, https://doi.org/10.5281/zenodo.8369670

6. De Maio, V., Aral, A., Brandic, I.: A roadmap to post-moore era for distributed systems. In: Proceedings of the 2022 Workshop on Advanced tools, programming languages, and PLatforms for Implementing and Evaluating algorithms for Distributed systems. pp. 30–34 (2022)
7. Farhi, E., Goldstone, J., Gutmann, S.: A quantum approximate optimization algorithm (2014)
8. Fu, X., Yu, J., Su, X., Jiang, H., Wu, H., Cheng, F., Deng, X., Zhang, J., Jin, L., Yang, Y., et al.: Quingo: A programming framework for heterogeneous quantum-classical computing with nisq features. ACM Transactions on Quantum Computing **2**(4), 1–37 (2021)
9. Gard, B.T., Motes, K.R., Olson, J.P., Rohde, P.P., Dowling, J.P.: An Introduction to Boson-Sampling, p. 167–192. WORLD SCIENTIFIC (Jun 2015). https://doi.org/10.1142/9789814678704_0008, http://dx.doi.org/10.1142/9789814678704_0008
10. Grover, L.K.: A fast quantum mechanical algorithm for database search. In: Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. pp. 212–219 (1996)
11. Hadfield, S., Wang, Z., O'Gorman, B., Rieffel, E., Venturelli, D., Biswas, R.: From the quantum approximate optimization algorithm to a quantum alternating operator ansatz. Algorithms **12**(2), 34 (Feb 2019). https://doi.org/10.3390/a12020034, http://dx.doi.org/10.3390/a12020034
12. Hevia, J.L., Peterssen, G., Piattini, M.: Quantumpath: A quantum software development platform. Software: Practice and Experience **52**(6), 1517–1530 (2022)
13. Kadowaki, T., Nishimori, H.: Quantum annealing in the transverse ising model. Physical Review E **58**(5), 5355–5363 (Nov 1998). https://doi.org/10.1103/physreve.58.5355, http://dx.doi.org/10.1103/PhysRevE.58.5355
14. King, A.D., Nocera, A., Rams, M.M., Dziarmaga, J., Wiersema, R., Bernoudy, W., Raymond, J., Kaushal, N., Heinsdorf, N., Harris, R., et al.: Computational supremacy in quantum simulation. arXiv preprint arXiv:2403.00910 (2024)
15. Lucas, A.: Ising formulations of many np problems. Frontiers in physics **2**, 74887 (2014)
16. Magann, A.B., Rudinger, K.M., Grace, M.D., Sarovar, M.: Feedback-based quantum optimization. Physical Review Letters **129**(25) (Dec 2022). https://doi.org/10.1103/physrevlett.129.250502, http://dx.doi.org/10.1103/PhysRevLett.129.250502
17. McCaskey, A.J., Lyakh, D.I., Dumitrescu, E.F., Powers, S.S., Humble, T.S.: Xacc: a system-level software infrastructure for heterogeneous quantum–classical computing. Quantum Science and Technology **5**(2), 024002 (2020)
18. McKay, D.C., Hincks, I., Pritchett, E.J., Carroll, M., Govia, L.C., Merkel, S.T.: Benchmarking quantum processor performance at scale. arXiv preprint arXiv:2311.05933 (2023)
19. Mintz, T.M., Mccaskey, A.J., Dumitrescu, E.F., Moore, S.V., Powers, S., Lougovski, P.: Qcor: A language extension specification for the heterogeneous quantum-classical model of computation. ACM Journal on Emerging Technologies in Computing Systems (JETC) **16**(2), 1–17 (2020)
20. Nielsen, M.A., Chuang, I.L.: Quantum computation and quantum information, vol. 2. Cambridge university press Cambridge (2001)
21. Piontek, T., Bosak, B., Ciżnicki, M., Grabowski, P., Kopta, P., Kulczewski, M., Szejnfeld, D., Kurowski, K.: Development of science gateways using qcg—lessons learned from the deployment on large scale distributed and hpc infrastructures. Journal of Grid Computing **14**, 559–573 (2016)

22. Shor, P.W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM review **41**(2), 303–332 (1999)
23. The CUDA Quantum development team: Cuda-q (2024), `https://github.com/NVIDIA/cuda-quantum`

Extended abstract in Polish

Zastosowanie metod obliczeń kwantowych do rozwiązywania problemów optymalizacyjnych.

1. Wprowadzenie

Informatyka kwantowa jest dziedziną nauki, w której, do wykonywania obliczeń, wykorzystuje się zjawiska kwantowe, takie jak superpozycja, splątanie i tunelowanie. Urządzenia, które przy zastosowaniu tych zjawisk są w stanie rozwiązywać uniwersalne problemy obliczeniowe, nazywane są komputerami kwantowymi. Podstawową jednostką informacji w komputerze kwantowym jest kubit, a podstawowymi elementami reprezentującymi możliwe do wykonania operacje na kubitach, są bramki kwantowe. Oprócz komputerów kwantowych istnieje wiele innych urządzeń kwantowych, które nie realizują w pełni założeń kwantowej maszyny Turinga, jednak znajdują zastosowania w specyficznych typach obliczeń. Do takich urządzeń zaliczamy m.in. kwantowe wyżarzacze oraz urządzenia realizujące próbkowanie bozonów.

Rozaważane w rozprawie problemy optymalizacyjne należą do problemów obliczeniowych, w których celem jest znalezienie rozwiązań najlepszych (rozwiazaniem optymalnych) wśród rozwiązań spełniających określone ograniczenia (rozwiązań dopuszczalnych). Oceny rozwiązań dokonuje się na podstawie ustalonych kryteriów oceny, najczęściej wyrażonych za pomocą odpowiednich funkcji. Poszukiwane rozwiązania problemów wygodnie jest reprezentować za pomocą wektora niewiadomych, czyli tzw.zmiennych decyzyjnych. W takich przypadku pojedyncze rozwiązanie stanowić będzie dowolne przypisanie wartości do tych zmiennych Problemy optymalizacyjne można klasyfikować na różne sposoby, m.in. według typu użytych zmiennych, które mogą być ciągłe lub dyskretne. Jeśli w modelu problemu konieczne jest użycie obu typów zmiennych, problem taki nazywany jest mieszanym. Typ użytych w modelu problemu zmiennych wpływa na liczność zbioru jego rozwiązań. Jeśli zbiór ten jest skończony lub policzalny, problem optymalizacyjny nazywamy kombinatorycznym. Ze względu na swoją złożoność, wiele problemów optymalizacyjnych, w tym kombinatorycznych, jest trudnych do rozwiązania przy użyciu współczesnych komputerów.

Informatyka kwantowa jest stosunkowo nową dziedziną nauki w porównaniu z informatyką klasyczną. Pierwsze komputery kwantowe z kilkoma kubitami powstały pod koniec lat 90. Konstruowanie komputerów kwantowych jest trudne ze względu na wyzwania inżynieryjne wynikające z mechaniki kwantowej. W efekcie, zastosowania obliczeń kwantowych do rozwiązywania problemów optymalizacyjnych są głównie teoretyczne i sprowadzają się najczęsciej do prostych implementacji na małej liczbie kubitów, w środowisku dodatkowo obciążonym zaszumieniem wynikającym z istniejących ograniczeń technologicznych. Celem tych realizacji jest pokazanie potencjału podejść

kwantowych, mimo problemów związanych z szumem i dekoherencją. Algorytmy kwantowe są probabilistyczne, co oznacza, że nie gwarantują znalezienia rozwiązań najlepszych (jeśli takowe istnieją), a jedynie zwracają je z określonym prawdopodobieństwem, zależnym od liczby pomiarów.

Jednym z dwóch najbardziej znanych algorytmów kwantowych jest bez wątpienia algorytm Grovera. Jest to jeden z pierwszych algorytmów, który wykazał potencjalną przewagę kwantową nad podejściami klasycznymi w rozwiązywaniu rzeczywistych problemów. Algorytm Grovera potrafi znaleźć element w nieustrukturyzowanej bazie danych szybciej (w stopniu wyrażonym za pomocą funkcji kwadratowej) w porównaniu do najlepszego algorytmu klasycznego. Dzięki swojej kwantowej przewadze i wszechstronności, algorytm Grovera często stosowany jest jako podprogram w bardziej złożonych algorytmach.

Mimo że obecne komputery kwantowe oferują ponad sto kubitów, co teoretycznie mogłoby pozwolić na wykazanie przewagi kwantowej nad komputerami klasycznymi dla wielu realnych problemów obliczeniowych, ich rzeczywiste możliwości są ograniczone do obliczeń na kilkudziesięciu kubitach i kilkudziesięciu warstwach kwantowych. Jest to znacznie mniej, niż potrzeba do skutecznego uruchomienia algorytmu Grovera, nie wspominając o jego rozszerzeniach. Algorytmy, które lepiej sprawdzają się na współczesnych komputerach kwantowych, to tzw. algorytmy wariacyjne, z których najbardziej znane to Variational Quantum Eigensolver (VQE) oraz Quantum Approximate Optimization Algorithm (QAOA). Oba są algorytmami hybrydowymi - kwantowo-klasycznymi.

Ze względu na ograniczenia współczesnych komputerów kwantowych, obecne badania i zastosowania obliczeń kwantowych w problemach optymalizacyjnych można sklasyfikować w następujący sposób:

- Badania aplikacyjne koncentrują się głównie na pogłębianiu zrozumienia znanych algorytmów służących do rozwiązywania standardowych problemów, takich jak MaxCut, które mają jedynie ograniczone zastosowania, np. w produkcji układów scalonych.

- Aplikacje skoncentrowane na zastosowaniu już wcześniej wypracowanych podejść dla nieuniwersalnych urządzeń kwantowych (takich jak komputerów realizujących kwantowe wyżarzanie) do kolejnych rodzajów problemów.

- Badania aplikacyjne, które przenoszą sformułowania opracowane dla nieuniwersalnych urządzeń kwantowych bezpośrednio do uniwersalnych paradygmatów dla wybranych problemów świata rzeczywistego, bez przeprowadzania dogłębnej analizy lub pełnego wykorzystania potencjału komputerów kwantowych,

Liczba publikacji dotyczących rozważanego tematu, które nie mieszczą się w jednej z tych trzech grup, jest bardzo ograniczona.

Równocześnie środowisko obejmujące takie elementy, jak badania naukowe, zastosowania przemysłowe, szkolenie nowych użytkowników, rozwój oprogramowania i integracja sprzętu, tworzy optymalne warunki dla długoterminowych postępów w danej dziedzinie. W przypadku obliczeń kwantowych zaobserwować można aktywny rozwój badań naukowych. Dobrze rozwinięte są również narzędzia do programowania na poziomie bramek kwantowych. Wciąż jednak istnieje duża potrzeba rozwoju zastosowań przemysłowych i wysokopoziomowych abstrakcji, w szczególności tych zintegrowanych z klasycznymi superkomputerami, aby umożliwić użytkownikom skupienie się na aplikacjach, zamiast na przyswajaniu nowinek technicznych i ich implementacji .

Niniejsza praca doktorska koncentruje się na wspomnianych wcześniej brakach w obszarze informatyki kwantowej, w szczególności w zastosowaniach uniwersalnych (bramkowych) komputerów kwantowych oraz na braku wysokopoziomowych abstrakcji. Temat pracy, czyli zastosowanie metod

obliczeń kwantowych do rozwiązywania problemów optymalizacyjnych, realizowany jest na kilka sposobów. Po pierwsze, skoncentrowano się na rozwiązaniu dobrze znanego problemu szeregowania, mającego szerokie zastosowanie w przemyśle. Dodatkowo wprowadzono i rozwiązano mniej znane problemy, głównie z wykorzystaniem algorytmu QAOA i jego wariantów, zarówno na symulatorach komputerów kwantowych, jak i na sprzęcie rzeczywistym. Niektóre wyniki porównano z wynikami uzyskanymi za pomocą komputerów realizujących kwantowe wyżarzanie. Praca nie ogranicza się jedynie do opracowania reprezentacji Hamiltonianowej służącej do rozwiązywania problemów przez algorytmy kwantowe, ale proponuje również nowe podejścia do ich rozwiązywania. Wypracowane aplikacje są dostarczane w postaci dwóch wyspecjalizowanych, lecz przyjaznych dla użytkownika platform, z których każda korzysta z modułowego narzędzia QCG-QuantumLauncher. Narzędzie to zaprojektowano do uruchamiania algorytmów kwantowych i rozwiązywania różnorodnych scenariuszy na różnych docelowych architekturach kwantowych.

2. Prace Badawcze

2.1 Efektywność algorytmu QAOA w problemach kombinatorycznych

Jak wykazano, algorytm QAOA z domyślnie sformułowanymi Hamiltonianami może być z powodzeniem stosowany do rozwiązywania problemów kombinatorycznych. Pokazano, że jakość wyników uzyskanych na symulatorze nie różni się znacząco od tych uzyskanych na rzeczywistym sprzęcie. Zarówno w przypadku symulatorów kwantowych, jak i rzeczywistych komputerów, rozmiary rozwiązywanych instancji są obecnie ograniczone do kilkudziesięciu zmiennych. Na przykład, odpowiada to rozwiązywaniu instancji z 5 pojazdami dla problemu ładowania pojazdów elektrycznych na autostradzie. Na komputerze ibmq_toronto możliwe jest osiągnięcie 0.5% prawdopodobieństwa znalezienia dopuszczalnego rozwiązania dla takich instancji. Te rozmiary instancji są znacząco mniejsze w porównaniu do tych, które mogą być obecnie rozwiązywane przez komputery realizujące kwantowe wyżarzanie, zdolne do rozwiązywania instancji zawierających aż 50 pojazdów elektrycznych. Podobnie, dla problemu taktycznej dekonfliktacji statków powietrznych, instancje o rozmiarze 5×4 mogą być rozwiązane z prawdopodobieństwem nie większym niż 0.01%.

Istotny okazał się dobór odpowiedniej głębokości algorytmu QAOA. Głębsze obwody (z większym parametrem p) lepiej przybliżają ewolucję adiabatyczną i mogą dawać lepsze wyniki. Jednak w przypadku rzeczywistego sprzętu kwantowego oraz symulatorów z szumem, głębsze obwody są bardziej podatne na szum, dekoherencję i interferencje. Dla problemu ładowania pojazdów elektrycznych wykazano, że najlepsze wyniki uzyskuje się przy obwodach o głębokości $p = 2$, ponieważ obwody o głębokości $p = 1$ nie są w stanie wystarczająco dobrze przybliżyć ewolucji adiabatycznej, a obwody o głębokości $p = 3$ lub większej nie są w stanie znaleźć dopuszczalnych rozwiązań. Warto jednak zauważyć, że wzrost głębokości obwodu nie wpływa znacząco na czas obliczeń potrzebnych do optymalizacji parametrów wariacyjnych; na przykład, czas obliczeń dla głębokości $p = 5$ jest tylko 1,43 razy dłuższy niż dla $p = 1$.

Kolejnym wnioskiem z eksperymentów nad algorytmem QAOA jest to, że kluczowe jest staranne sformułowanie Hamiltonianu kosztu, ponieważ prawdopodobieństwo sukcesu może znacznie się różnić w zależności od struktury instancji, nawet jeśli jej rozmiar pozostaje ten sam. Eksperymenty dla instancji problemu dekonfliktacji statków powietrznych o rozmiarze 12 zmiennych pokazały, że łatwiej jest znaleźć dopuszczalne rozwiązania, gdy jest więcej statków powietrznych, a mniej manewrów. Z kolei, gdy jest mniej statków powietrznych i więcej manewrów, prawdopodobieństwo sukcesu spada od 2 do 3 razy, mimo że iloczyn tych dwóch wartości pozostaje taki sam. Zmiana z problemów decyzyjnych na optymalizacyjne nie wydaje się natomiast zwiększać trudności w

rozwiązywaniu problemu. Jest to zgodne z oczekiwaniami, ponieważ Hamiltoniany dla problemów decyzyjnych i optymalizacyjnych różnią się tylko w termach jednokubitowych, nie wprowadzając dodatkowych splątań.

2.2 Połączenie algorytmu QAOA z klasycznymi zasobami superkomputerowymi

Związek między QAOA a ewolucją adiabatyczną sugeruje, że sekwencje parametrów wariacyjnych powinny monotonicznie wzrastać lub maleć wraz z głębokością obwodu. W rozprawie przebadano możliwość wykorzystania zasobów superkomputerów dostępnych w Poznańskim Centrum Superkomputerowym i Sieciowym do znalezienia optymalnych parametrów wariacyjnych dla problemu JSSP. W eksperyymentacg zaobserowano $6,5$-krotne skrócenie czasu optymalizacji między głębokościami $p = 3$ a $p = 4$. Efekt taki można uzyskać pod warunkiem wykorzystania optymalnych parametrów znalezionych dla obwodu QAOA o głębokości $p = 3$ jako punktów początkowych dla głębokości $p = 4$.

Warto również wspomnieć o związku między wartością energii a jakością rozwiązania w problemach optymalizacyjnych. Podobnie jak w sztucznej inteligencji, gdzie minimalizowana jest funkcja kosztu, w optymalizacji kwantowej minimalizowana jest energia. Wyniki pokazują, że w niektórych przypadkach rozwiązania niedopuszczalne mogą mieć energię niższą niż rozwiązania dopuszczalne, w szczególności w kontekście optymalizacji całkowitego czasu uszeregowania. Niemniej jednak ogólna zasada, że niższa energia odpowiada wyższemu prawdopodobieństwu uzyskania wysokiej jakości rozwiązań, pozostaje aktualna. W toku badań stwierdzono istotny potencjał dostosowywania parametrów decyzyjnych, takich jak maksymalny czas, w celu poprawy wyników.

2.3 Optymalizacja wykorzystująca jedynie obliczenia kwantowe

Podejście znajdowania optymalnych parametrów wariacyjnych jest kosztowne zarówno pod względem zasobów obliczeniowych, jak i całkowitego czasu obliczeń. Z drugiej strony, istnieje np. algorytm FALQON, którego istotą jest możliwość pominięcia klasycznej części obliczeń.

Wyniki przeprowadzonych eksperymentów wskazują, że algorytm FALQON może osiągać znacznie niższe poziomy energii w porównaniu do standardowego QAOA. Ważne jest jednak to, że podczas gdy liczba warstw QAOA pozostaje stała w trakcie obliczeń, FALQON wymaga iteracyjnego pogłębiania obwodu, co zwiększa prawdopodobieństwo ingerencji szumu w uzyskane rezultaty. Wyniki uzyskano przy użyciu symulatorów idealnych, co oznacza, że należy je traktować jako ciekawą inspirację do dalszych badań w przyszłości, a nie jako praktyczne podejście do wykorzystania na obecnych komputerach kwantowych.

2.4 Redukcja przestrzeni poszukiwań

Kolejnym sposobem, który pozwala na dalszą poprawę wyników w optymalizacji kwantowej, jest podejście oparte na włączeniu twardych ograniczeń w Hamiltonianie miksującym za pomocą algorytmu QAOA Ansatz. To podejście redukuje przestrzeń poszukiwań, zwiększając prawdopodobieństwo znalezienia rozwiązania dopuszczalnego. Odbywa się to jednak kosztem dodania większej liczby bramek dwukubitowych. Nie wszystkie twarde ograniczenia muszą być włączone do Hamiltonianu miksującego, ponieważ nie dla wszystkich ograniczeń jest to korzystne. W przypadku problemów decyzyjnych zazwyczaj nie można uwzględnić wszystkich ograniczeń. Jednak w

odniesieniu do problemów omawianych w pracy udało się wyodrębnić podzbiór ograniczeń, które włączone do Hamiltonianu miksera, poprawiły skuteczność algorytmu.

W przypadku problemu taktycznej dekonflikcji statków powietrznych, włączenie twardego ograniczenia mówiącego, że każdy statek wykonuje dokładnie jeden manewr, do Hamiltonianu miksera zwiększyło prawdopodobieństwo pomiaru rozwiązania dopuszczalnego w instancji z 5 statkami powietrznymi, z których każdy miał 3 alternatywne manewry, z 11,69% do 77,33%, co stanowi ponad sześciokrotny wzrost. Jednakże to podejście nie poprawiło znacząco ogólnego kształtu rozkładu prawdopodobieństwa w optymalizacji mającej na celu minimalizację całkowitej liczby zmian wymaganych w pierwotnym planie lotów.

2.5 Złagodzenie ograniczeń w formułowaniu Hamiltonianów

Nieintuicyjne wydawać się może, że przyjęcie odwrotnego podejścia do ograniczenia przestrzeni poszukiwań może przynieść podobne rezultaty. Wykazano, że złagodzenie sformułowania problemu, aby uwzględnić szerszy zakres rozwiązań jako dopuszczalne, przy jednoczesnym zmniejszeniu liczby splątań, również wydaje się prowadzić do lepszych wyników. Jest to nowe podejście, opisane po raz pierwszy w niniejszej pracy doktorskiej. Polega ono na uproszczeniu Hamiltonianów generujących obwody posiadające wiele bramek dwukubitowych. Oznacza to, że w modelu problemu taktycznej dekonflikatcji dopuszcza się w danej chwili wykonanie przez statek powietrznych większej (niż standardowo - jeden) liczby manewrów jednocześnie. Dzięki tej modyfikacji zaobserwowano poprawę wydajności nie tylko w porównaniu do standardowych sformułowań Hamiltonianu, ale także do sformułowań kwadratowych (QUBO) znanych z komputerów opartych o kwantowe wyżarzanie, które obejmują najwyżej splątania drugiego stopnia. Przykładowo, w scenariuszu z 3 statkami powietrznymi i 5 manewrami zarejestrowano prawdopodobieństwo zmierzenia dopuszczalnego rozwiązania wynoszące $0,0009\%$, w porównaniu do $0,0007\%$ dla podejścia QUBO i $0,0001\%$ dla standardowego sformułowania, w szczególnie trudnej instancji, gdzie istnieje tylko 5 wykonalnych rozwiązań spośród 215. Co ciekawe, wyniki te sugerują, że w przyszłości uniwersalne obliczenia kwantowe mogą stać się bardziej efektywne niż realizacje algorytmów kwantowego wyżarzania.

3. Prace wdrożeniowe

3.1 Portal QCG jako webowe narzędzie do rozwiązywania problemów kombinatorycznych na komputerach kwantowych i symulatorach

Poznańskie Centrum Superkomputerowo-Sieciowe, jako członek konsorcjum, brało udział w projekcie EuroHPC-PL, którego celem było zbudowanie krajowej infrastruktury do obliczeń dużej skali dla badań i przemysłu. PCSS było odpowiedzialne za opracowanie platformy do kwantowych badań operacyjnych i optymalizacji dyskretnej, a także za uzyskanie dostępu do infrastruktury do kwantowej optymalizacji kombinatorycznej. Pierwszy z tych elementów, o poziomie gotowości technologicznej (TRL) 9, był centralnym punktem badań wdrożeniowych w tej pracy. Cele projektu zrealizowano poprzez rozszerzenie i integrację istniejącego zestawu oprogramowania QCG oraz opracowanie nowych, nieistniejących wcześniej komponentów.

Podczas realizacji projektu stworzono warstwę dostępową do osadzania szablonów aplikacji oraz aplikacji zorientowanych na problem, oferując elastyczną i dostosowywalną platformę do różnych zastosowań. Platforma ta została zbudowana na komponencie QCG-Portal, umożliwiając użytkownikom przeglądanie, kontrolowanie i monitorowanie zadań za pomocą interfejsu webowego. Została ona również zintegrowana z QCG-Templates, co pozwoliło na dostosowanie widoków

aplikacji poprzez parametryzację. Ponadto, wykorzystano QCG-API i QCG-Agent do autoryzacji i przesyłania zadań.

Opracowana platforma zawiera kilka wstępnie zdefiniowanych aplikacji graficznych, które pozwalają użytkownikom definiować instancje wybranych i zaimplementowanych problemów optymalizacji kombinatorycznej, takich jak problem MaxCut, problem Exact Cover oraz JSSP. Interfejs umożliwia użytkownikom określenie rozmiaru i rodzaju instancji za pomocą intuicyjnego interfejsu graficznego lub tekstowego, bądź przez wczytanie wcześniej istniejących danych z pliku. Platforma wspiera również przesyłanie ogólnych zadań obliczeniowych dla dowolnego problemu optymalizacji poprzez wprowadzenie macierzy QUBO w interfejsie tekstowym lub przez wgranie jej z pliku. Użytkownicy mogą następnie wybrać architekturę, na której problem zostanie wykonany, w tym rzeczywiste komputery kwantowe lub symulatory wdrożone podczas projektu PRACE-LAB2. Dodatkowe opcje umożliwiają wybór odpowiedniego algorytmu optymalizacyjnego oraz dostosowanie jego hiperparametrów. Warto również zauważyć, że klasyczne części algorytmów hybrydowych zostały przygotowane do równoległego przetwarzania, co umożliwia ich wykonywanie na klastrach HPC lub superkomputerach, znacząco przyspieszając obliczenia.

Aplikacje dla wybranych problemów optymalizacji dyskretnej są dostępne za pośrednictwem QCG-Portal, który został zaadaptowany i funkcjonalnie rozszerzony na potrzeby projektu. QCG-Portal służy również jako narzędzie do zarządzania zasobami, uwierzytelniania użytkowników, przesyłania zadań do systemu kolejkowania oraz pobierania metadanych dotyczących przesłanych zadań. Opracowane szablony w ramach platformy ułatwiają również analizę wyników zadań za pomocą interaktywnego interfejsu graficznego, który prezentuje wyniki w formie wykresów dostosowanych do konkretnego problemu, algorytmu oraz architektury kwantowej. Należy zauważyć, że komponenty QCG wymagały dodatkowego rozszerzenia, aby w pełni obsługiwać przypadki użycia związane z obliczeniami kwantowymi. Konieczne było także wdrożenie widoków specyficznych dla tych zastosowań. Jednym z zaawansowanych narzędzi, które musiały zostać opracowane od podstaw, był QCG-QuantumLauncher (QCG-QL). Biblioteka ta umożliwia łatwe wykonywanie problemów kombinatorycznych na różnych komputerach kwantowych z użyciem wybranych algorytmów kwantowych.

3.2 Skydodge: menadżer ruchu lotniczego

Projekt EuroHPC PL skupiał się na dostarczeniu użytkownikom narzędzia do rozwiązywania wybranych problemów optymalizacji kombinatorycznej lub własnych macierzy QUBO na maszynach kwantowych. Chociaż te rozwiązania zostały pomyślnie wdrożone w ramach projektu, w niektórych scenariuszach wymagana była bardziej szczegółowa i specyficzna integracja z istniejącymi procesami. Tak było w przypadku projektu Quantum Air Traffic Management (QATM) (TRL 6), gdzie opracowano rozwiązanie kwantowe wspierające kontrolerów ruchu lotniczego.

Produkt opracowany w ramach projektu składa się z trzech modułów: logicznej warstwy kwantowej, która przetwarza dane wejściowe dotyczące statków powierznych i generuje rozwiązanie problemu, QCG-Template, dostępnego przez QCG-Portal -umożliwiającego uruchamianie obliczeń na sprzęcie kwantowym lub klasycznych symulatorach, oraz SkyDodge - interaktywnego interfejsu monitorującego sytuację w przestrzeni powietrznej i wizualizującego potencjalne rozwiązania znalezione przez maszyny kwantowe. Kwantowa warstwa logiczna opiera się na efektach badań zebranych w opublikowanych artykułach i podsumowanych we wcześniejszych punktach.

Moduł kwantowy składa się z dwóch podmodułów: podmodułu generatora oraz podmodułu dekonfliktacji. Te podmoduły są zintegrowane na zasadzie działania w pętli. Generator tworzy scenariusze ruchu lotniczego z konfliktami i generuje możliwe trasy dla samolotów. Te trasy są

następnie przesyłane do podmodułu dekonfliktacji, który identyfikuje konflikty i wykorzystuje sprzęt kwantowy do wyboru tras bezkonfliktowych, spełniających dodatkowe ograniczenia. Proces powtarza się, a generator aktualizuje scenariusze, dodaje sytuacje losowe (między innymi takie jak burze) i generuje nowe trasy, aż do zakończenia symulacji. Podmoduł dekonfliktacji umożliwia także przypisywanie wag do tras na podstawie preferencji użytkownika, co pozwala na optymalizację, taką jak chociażby minimalizacja zużycia paliwa lub priorytetyzacja określonych lotów. Takie podejście zapewnia dużą elastyczność w zarządzaniu sytuacjami w przestrzeni powietrznej i umożliwia analizę oraz rozwiązywanie konfliktów w czasie bliskim rzeczywistemu.

Dobre praktyki ustalone podczas pracy nad projektem EuroHPC PL ułatwiły efektywne zaprojektowanie i wdrożenie widoku QCG-Portal. Interfejs ten pozwala użytkownikom na przesyłanie zadań na różne architektury kwantowe, w tym architekturę bramkowaą. Ponadto użytkownicy mają możliwość generowania instancji z wybraną liczbą statków powietrznych i manewrów oraz doboru parametrów sterowania, takich jak czas pętli odpowiedzi. System wizualizacji, SkyDodge, działa w trybie odtwarzania i przetwarza cały scenariusz ruchu lotniczego aż do jego zakończenia. System potrafi wizualizować trasy statków powietrznych, oznaczać sytuacje konfliktowe, wyświetlać wybrane alternatywne trasy, dostosowywać prędkość odtwarzania, przewijać scenariusz od wybranego momentu, filtrować podzbiory statków powietrznych oraz prezentować dodatkowe informacje o locie i warunkach powietrznych. Został on zaprojektowany w taki sposób, aby jasno pokazać działania kontrolera lub systemu dekonfliktacji oraz umożliwić szczegółową analizę podjętych decyzji.

3.3 QCG-QuantumLauncher: modułowy zestaw narzędzi do zarządzania kwantowymi scenariuszami aplikacyjnymi

Wstępne eksperymenty mające na celu zbadanie możliwości technologii kwantowych ujawniły znaczącą lukę w rozwiązaniach programistycznych wysokiego poziomu, które ułatwiłyby przeprowadzanie takich eksperymentów. Po pierwsze, oprócz podstawowych zasad formułowania Hamiltonianów w problemach optymalizacyjnych, nie istniało żadne narzędzie, które automatyzowałoby ten proces, co wymagało ręcznego tłumaczenia funkcji logicznych na Hamiltoniany. W związku z tym zdecydowano się opracować wewnętrzne narzędzie do tłumaczenia funkcji logicznych na Hamiltoniany, nazwane QCG-Hampy. Po drugie, dla każdego kolejnego podejścia eksperymentalnego cała ścieżka przetwarzania, w tym odczyt danych, wybór backendu kwantowego, wybór algorytmu, zapisywanie wyników itp., musiała być ponownie implementowana w kodzie. Aby uprościć ten proces, opracowano bibliotekę programistyczną o nazwie QCG-QuantumLauncher (QCG-QL), której QCG-Hampy został częścią, a która przekształciła się w dojrzałe i modułowe rozwiązanie do rozwiązywania klasycznych problemów obliczeniowych przy użyciu algorytmów kwantowych na komputerach kwantowych.

Główną korzyścią płynącą z użycia QCG-QL, którą można zaobserwować na pierwszy rzut oka, jest znaczące uproszczenie, jakie oferuje w zakresie wykonywania algorytmów kwantowych do rozwiązywania konkretnych problemów na różnych maszynach kwantowych. Jedną z zauważalnych zalet jest znaczne zmniejszenie wymaganej liczby linii kodu. W typowych eksperymentach QCG-QL redukuje kod z kilkuset linii do zaledwie kilku. Przykładowo, typowa implementacja QAOA rozwiązująca JSSP na bramkowym komputerze kwantowym, w tym proste zapisywanie wyników, zazwyczaj wymaga około 400 linii kodu. Używając QCG-QL, wystarczą jedynie 5 linii kodu. Redukcja ta nie byłaby tak imponująca, gdyby QCG-QL jedynie opakowywał konkretne bloki kodu w funkcje lub klasy. Jednak QCG-QL oferuje pełną modularność i elastyczność. Jego kluczową zaletą w porównaniu do podobnych bibliotek oprogramowania kwantowego jest możliwość przełączania

się między różnymi architekturami kwantowymi. Użytkownicy mogą przełączyć się z bramkowego komputera kwantowego na komputer oparty o kwantowe wyżarzanie lub komputer realizujący próbkowanie bozoniczne za pomocą zaledwie jednej linii kodu. To samo dotyczy przełączania między róznymi problemami optymalizacyjnymi lub użytymi do ich rozwiązywania algorytmami. Wszystkie wymagane procesy obsługiwane są przez QCG-QL w sposób niewidoczny dla użytkownika, przy minimalnym jego zaangażowaniu.

Chociaż narzędzie zostało po raz pierwszy publicznie zaprezentowane dopiero niedawno podczas wystąpienia konferencyjnego, było ono już wcześniej wykorzystywane do celów wewnętrznych. QCG-QL okazało się nieocenione w przeprowadzaniu eksperymentów dla prac zajmujących się problemem taktycznej dekonfliktacji statków powietrznych a także problemem ładowania pojazdów z silnikiem elektrycznym. QCG-QL służył także jako komponent kwantowy dla pełnych produktów i wyników projektów QATM i EuroHPC PL. Dzięki swojej dużej elastyczności i rozszerzalności QCG-QL umożliwił wdrożenie koncepcji integracji superkomputerów z komputerami kwantowymi w celu opracowania algorytmu hybrydowego, opisanego w następnym punkcie. Ponadto QCG-QL stanowi fundament dla przyszłych projektów integracyjnych. Biorąc pod uwagę wymienione zastosowania w wielu obszarach, QCG-QL można uznać za skuteczne narzędzie wspierające zastosowanie metod obliczeń kwantowych w rozwiązywaniu problemów optymalizacyjnych.

3.4 Rozszerzenia QCG-QL do rozwiązywania problemów kombinatorycznych w systemach hybrydowych

Jednym ze szczególnie istotnych obszarów badań z zakresu obliczeń kwanotwych jest rozwój hybrydowych algorytmów klasyczno-kwantowych. Mało prawdopodobne jest, aby komputery kwantowe realizowały obliczenia niezależnie. Bardziej prawdopodobnym scenariuszem jest synergiczne współdziałanie komputerów klasycznych i kwantowych. Spodziewać się należy, że komputery kwantowe będą wykonywały pewne specyficzne obliczenia, przyspieszając obliczenia klasyczne, ale również obliczenia klasyczne będą stanowiły wsparcie dla obliczeń kwantowych.

W przypadku tego specyficznego, lecz szerokiego obszaru hybrydowych obliczeń klasyczno-kwantowych, QCG-QL również wykazuje swoją przydatność. W połączeniu z narzędziem QCG-PilotJob (QCG-PJ), usługą QCG zaprojektowaną do efektywnego wykonywania wielu zadań w ramach jednej alokacji, QCG-QL został wykorzystany do opracowania hybrydowego algorytmu, który wykorzystuje zasoby superkomputerowe do przyspieszenia procesu znajdowania optymalnych parametrów wariacyjnych dla algorytmu QAOA. Znalezienie optymalnych parametrów wariacyjnych dla QAOA jest trudne i kosztowne obliczeniowo. W omawianym podejściu narzut obliczeniowy jest nierównomierny, ponieważ większość obliczeń koncentruje się na początkowym etapie. To podejście ma również istotną wadę, polegającą na konieczności ustalenia liczby punktów początkowych podczas inicjalizacji algorytmu. Może to prowadzić do nadmiarowych obliczeń, jeśli optymalny zestaw parametrów zostanie znaleziony wcześniej. Z drugiej strony, jeśli liczba punktów początkowych okaże się niewystarczająca, algorytm może nie znaleźć optymalnych parametrów wariacyjnych.

Nowe podejście zapewnia efektywne wykorzystanie zasobów w ramach alokacji, ponieważ głębsze obwody kwantowe zazwyczaj wymagają większej mocy obliczeniowej niż płytsze. Należy zauważyć, że każdy blok optymalizujący parametry wariacyjne, reprezentuje hybrydową klasyczno-kwantową optymalizację QAOA. Ponieważ oczekuje się, że obliczenia kwantowe będą znacznie szybsze niż klasyczne, dostęp do kilku zasobów kwantowych wystarczy aby uruchomić dziesiątki takich procesów. Komputer kwantowy może być współdzielony między procesami klasycznymi i

nie pozostaje bezczynny podczas klasycznej optymalizacji, co miałoby miejsce, gdyby każdy zasób kwantowy był powiązany z jedną jednostką klasyczną.

4. Podsumowanie

Niniejsza dysertacja podsumowuje prace przeprowadzone w trakcie czteroletnich studiów doktoranckich. Rozprawa rozpoczyna się od przeglądu obecnego stanu wiedzy w przedmitwej dziedzinie badań oraz identyfikacji kluczowych problemów. Kolejne badania doprowadziły do opracowania specjalistycznych narzędzi, które następnie wykorzystano do przyspieszenia dalszych prac badawczych, tworząc pętlę, która umożliwiła realizację praktycznych aplikacji. Mimo że nie wszystkie wyzwania w tej dziedzinie zostały rozwiązane, praca ta wnosi istotny, spójny i uporządkowany wkład, zwłaszcza w zakresie wiedzy naukowej, narzędzi wspierających użytkowników oraz aplikacji do optymalizacji kwantowej.

Zastosowoanie komputerów kwantowych w praktycznych zastosowaniach do rozwiązywania problemów optymalizacyjnych jest wciąż na początkowym etapie rozwoju. Większość obecnych rozwiązań dotyczy problemów o ograniczonym zastosowaniu. Problem ten wynika częściowo z nowości samej technologii, ale także z wysokiego progu wejścia dla specjalistów z przemysłu, spowodowanego brakiem narzędzi o wysokim poziomie abstrakcji, które ułatwiłyby przeprowadzanie obliczeń. Dodatkowo sytuację komplikuje istnienie różnych paradygmatów kwantowych, które wymagają podejść dedykowanych.

Pierwszą luką w dotychczasowych badaniach, którą zidentyfikowano i zajęto się w tej pracy, były eksperymenty z algorytmem QAOA. Sformułowano dedykowane Hamiltoniany dla JSSP, problemu ładowania pojazdów elektrycznych i problemu taktycznej dekonflikcji. Przeprowadzono eksperymenty dotyczące QAOA, obejmujące analizę strategii interpolacji optymalnych parametrów wariacyjnych, porównanie z kwantowym wyżarzaniem oraz badanie związku między energią a długością znalezionego uszeregowania zadań. Eksperymenty te pokazały praktyczne zastosowanie komputerów kwantowych do rozwiązywania rzeczywistych problemów optymalizacyjnych, a także przyczyniły się do wczesnego wykorzystania narzędzia QCG-QL oraz opracowania warstwy logicznej dla portalu EuroHPC PL i pomyślnego zakończenia projektu QATM.

Podczas prac badano także rozszerzenia algorytmów QAOA. Czysto kwantowa optymalizacja z użyciem algorytmu FALQON radziła sobie lepiej niż QAOA, szczególnie w przypadku długich obwodów, które jednak nie są możliwe do realizacji na obecnych komputerach kwantowych. Algorytm ograniczający przestrzeń poszukiwań okazał się obiecującą alternatywą, wykazując lepszą skuteczność w rozwiązywaniu problemów. Z kolei całkowicie nowe podejście, stojące w opozycji do obecnych, znanych algorytmów, a polegające na eksploracji większej liczby rozwiązań, także okazało się lepsze od standardowej wersji algorytmu QAOA. Co więcej, okazało się przewyższać tradycyjną metodę QUBO stosowaną na komputerach realizujących kwantowe wyżarzanie. Prowadzenie szerokich eksperymentów w ramach pracy było znacznie ułatwione dzięki wcześniej zaimplementowanym narzędziom, takim jak QCG-Hampy i QCG-QL.

Prace badawcze były ściśle powiązane z rozwojem aplikacji, co stworzyło samonapędzającą się pętlę. Główną siłą napędową tej pętli było narzędzie QCG-QL, które powstało z potrzeby bardziej efektywnych badań, a ostatecznie stało się przedmiotem zainteresowania szerszego grona odbiorców. W pierwszej kolejności, QCG-QL rozwiązało problem powtarzalności kodu przy przeprowadzaniu wielu eksperymentów z różnymi problemami, algorytmami lub komputerami kwantowymi. Po drugie, QCG-QL dostarczyło warstwę wysokopoziomowej abstrakcji, ukrywając przed użytkownikami szczegóły implementacyjne, pozostawiając jednocześnie otwartą możliwość

dla bardziej zaawansowanych badaczy, którzy mogą rozszerzać moduły bazowe QCG-QL w celu prowadzenia bardziej zaawansowanych badań.

QCG-QL stanowiło kwantową bazę logiczno-obliczeniową dla dwóch projektów: EuroHPC PL i QATM. Projekt EuroHPC PL zakończył się zaprojektowaniem i wdrożeniem platformy do rozwiązywania problemów optymalizacji kombinatorycznej, która umożliwia użytkownikom korzystanie z graficznego interfejsu do definiowania zadań oraz łatwego dostępu do dostępnych komputerów kwantowych. Platforma została zintegrowana z narzędziem QCG-Portal, co pozwoliło na organizację zadań, analizę wyników oraz monitoring.

Projekt QATM był drugim projektem, który zakończył się sukcesem, w ramach którego QCG-QL stanowił bazę logiczno-obliczeniową. Rozwiązania wypracowane w pracach badawczych umożliwiły taktyczną dekonfliktację statków powietrznych za pomocą algorytmów kwantowych, z możliwością optymalizacji kryteriów, takich jak minimalizacja zużycia paliwa czy liczba zmian trajektorii dla lotów o wysokim priorytecie. Wzięto pod uwagę również czynniki środowiskowe, takie jak burze, oraz ograniczenia administracyjne. Kwantowe rozwiązanie zostało wdrożone w usłudze QCG-Portal, stanowiąc naturalne rozszerzenie rozwiązań opracowanych w ramach projektu EuroHPC PL. Dodatkowo opracowano specjalne narzędzie o nazwie SkyDodge do wizualnej analizy procesu dekonfliktacji w trybie odtwarzania, które może ostatecznie służyć jako wsparcie dla kontrolerów lotów.

© 2024 Tomasz Pecyna

Poznan University of Technology
Faculty of Computing
Institute of Computing Science

Typeset using LaTeX in Computer Modern.

BibTeX:

```
@mastersthesis{ key,
    author = "Tomasz Pecyna",
    title = "{application of quantum computing approaches for solving optimization problems}",
    school = "Poznan University of Technology",
    address = "Pozna{\'n}, Poland",
    year = "2024",
}
```