Quantum Computing for Earth Observation

Use cases timeline report

Written by	***Responsibility*** + handwritten signature if no electronic workflow tool
TASF Team *TRT Team*	Benjamin MARCHAND Noé OLIVIER
Verified by	
INFN Team	Dr. Ilaria SILOI, Marco TESORO
FZJ Team	Amer DELILBASIC, Edoardo PASETTO
IQM Team	Dr. Jay NATH, Dr. Jiri Guth JARKOVSKY, Dr. Markus WEBER
Approved by	

Approval evidence is kept within the document management system

Change Records

ISSUE	DATE	§ CHANGE RECORDS	AUTHOR
0	05/07/2023	Draft issue	TAS-F Team
0.1	10/08/2023	PM with ESA on August 10th	TAS-F & TRT Team
0.2	07/09/2023	Timeline description for use case n°1b and n°2	TAS-F & TRT Team
1.0	26/09/2023	Add Timeline description for use case n°1a, n°3 and n°4	INFN, TAS-I, FZJ teams
1.1	18/11/2023	Issue resolution, minor changes	FZJ Team

Table of contents

1 Introduction

1.1 Scope and context of the document

This document is part of a comparative study between existing quantum computers technologies. The study is particularly focused on integrated systems and Quantum Computing machines.

Based on the machines roadmaps defined in the previous phase of the study (WP3), we will derive a feasible timeline to process each of the identified Earth Observation use cases. Hence, regarding the algorithms that will run on the chosen hardware, we will estimate the minimum amount of needed resources, which is essential in the timeline definition.

Then we will estimate the level of investment that companies and research institutions can provide to make the proof of concept. Each investment will strongly rely on the capacity of each actor to have access to real hardware.

1.2 Applicable documents

[AD-1]	QC4EO Study Statement of Work
[AD-2]	Proposal submitted for QC4EO

1.3 Reference documents

[RD-1]	QC4EO-D1-Use Case Definition and Design Report
[RD-2]	QC4EO-D2-Machine Definition Report
[RD-3]	QC4EO-D3-Machines Roadmap Assessment Report

1.4 Acronyms

AR	Acquisition Request
DLO	Downlink Opportunity
DTO	Data Take Opportunity
EO	Earth Observation
FTQC	Fault-Tolerant Quantum Computing
KPI	Key Performance Indicator
ML	Machine Learning
NISQ	Noisy Intermediate Scale Quantum
PPO	Proximal Policy Optimization
PQC	Parametrized Quantum Circuits
QAOA	Quantum Approximate Optimization Algorithm
QC4EO	Quantum Computing For Earth Observation
QEC	Quantum Error Correction
QFT	Quantum Fourier Transform

QNN	Quantum Neural Network
QPU	Quantum Processing Unit
QTRL	Quantum Technology Readiness Level
QUBO	Quadratic Unconstrained Binary Optimization
RBF	Radial Basis Function
RDA	Range Doppler Algorithm
RL	Reinforcement Learning
SIFT	Scale Invariant Feature Transform
SotA	State-of-the-Art
SVM	Support Vector Machine
UC	Use Case
WP	Work Package

2 Use cases timeline structuring

This section describes the information used to operate the feasible timeline for each of the identified use cases:

- **Current performances of the selected algorithm:** Brief remind of algorithm and its actual performances on quantum machine (if any)

- **High-level description of algorithms implementation:** Description of the structure of the algorithm and highlight of the quantum part. When possible, distinction between the different subroutines of the quantum part ("algorithmic blocks")

- **Bottlenecks:** Bottlenecks in an algorithmic point of view (i.e. KPIs of WP2) and of different « algorithmic blocks » if several are existing

- **Original kernel simplification:** Identification of bottlenecks that necessitate to simplify the original kernel to a Noisy Intermediate Scale Quantum (NISQ) Computing

- **Hardware machines and roadmaps:** Comparative study (from WP3 results and bottlenecks) of different quantum machines types, and selection of the most appropriate for each use-case, considering its algorithm constraints defined in the previous section

- **Scaling of the proposed algorithms**: Study the scaling of the proposed algorithms on the selected machines

- **Cost of computing**: Highlight the resources required to solve a problem of a given size and how they scale with the problem size. One possible way (not exclusive) is, for complex algorithms that can be split in multiple "blocks", to establish when these different algorithmic blocks can be efficiently implemented. Idem for the simplified versions of the original kernel if such a simplification has been proposed

- **Expected time availability**: Establish an approximate timeline taking into account both a pessimistic and an optimistic development of the hardware (from WP3). This timeline must include expectations about when a minimal size problem and a full size problem can be tackled by the quantum approach

- **Bibliography:** references to the specific use case

3 List of use cases

The following table lists the use cases which have been already selected and analyzed in the report delivered by WP1. References to the corresponding timeline definition are provided.

USE CASE NUMBER	USE CASE TITLE	TIMELINE DESCRIPTION
1	Mission Planning for EO Acquisitions *Scenario 1a: Quantum annealing* *Scenario 1b: Quantum machine learning*	 See section 4.1.1 See section 4.1.2
2	Multiple-view Geometry on Optical Images	See section 4.2
3	Optical Satellite Data Analysis	See section 4.3
4	SAR Raw Data Processing	See section 4.4

4 Use cases timeline definition

4.1 Use case n°1: Mission Planning for EO Acquisitions

4.1.1 Scenario 1a: Quantum annealing

4.1.1.1 Proposed algorithms

4.1.1.1.1 Current performances of the selected algorithms

For the solution of the mission planning problem, we chose quantum annealing (QA) and its discretized version, the gate-model algorithm (*Quantum Approximate Optimization Algorithm*, QAOA). These approaches are heuristic methods that yield a candidate solution which is not guaranteed to be the exact solution. Strict bounds for the approximation ratio (how close is the found solution to the best possible solution) are not available. However, this is also true to operationally employed classical heuristics. An upper bound for a potential speed-up is assumed to be quadratic (cf. Grover algorithm). Current experiments are inconclusive concerning a potential quantum advantage as it is inherently difficult to compare heuristic method as the performance heavily depends on the specific problem instance to solve.

4.1.1.1.2 High-level description of algorithms implementation

Quantum annealing (QA) [Edward2000] and QAOA [Fahri2014] both aim to find the solution of a binary optimization problem. In current hardware this is limited to quadratic cost functions (*Quadratic Unconstrained Binary Optimization*, QUBO). The algorithms start with an equal superposition of all possible binary strings in a quantum register. Then the algorithmic primitives drive the quantum register towards the solution of the binary optimization problem. The latter is encoded into the quadratic couplings Q_{ij} and linear coefficients Q_{ii} of the QUBO problem $\sum_{ij} Q_{ij} x_i x_j, x_i \in \{0,1\}$. To solve the problem for quantum annealing or QAOA one needs to give the quadratic and linear coefficients, as well as multiple method specific parameters. For QA these are for example the annealing time and the initial state. For QAOA one needs to specify the number of layers, which roughly translate into the number of discretization steps in the QA-QAOA analogy. Another important variable is the choice of the initial state and the so-called driver (or mixer) operation.

4.1.1.1.3 Bottlenecks

The main bottlenecks for both QA and QAOA are:

1. **Precision**: The problem specification is limited by the machine precision of the couplings. This can lead to severe suppression of the performance [Stollenwerk2020] as it leads to a misspecification of the problem to be solved. For QAOA specifically, high error rates lead to limited usability for current NISQ devices due to quickly accumulating errors for large problem sizes (see also the discussion in Sec. 4.1.1.2.1.1)
2. **Connectivity**: Hard optimization problems like the mission planning problem at hand are characterized by nontrivial interactions. i.e., the non-vanishing quadratic couplings in the QUBO cannot be directly mapped to a two-dimensional graph as they occur in most solid-state based quantum computing platforms (superconducting, spin-qubits). In such cases, compilation strategies (embedding for QA, qubit routing for QAOA) must be employed, which may lead to a polynomial overhead in terms of quantum computing resources [Zhou2020].

3. **Limited width and depth**: The size of the problems that can be solved with QA and QAOA are limited by the number of qubits (circuit width) as well as the efficient circuit depth, which scales with the problem size. The latter is limited by decoherence and error rates as it is shown in the figures below.

4.1.1.1.3.1 Original kernel simplification for NISQ applications

This does not apply to the current use case, because we are not considering problem subdivision here.

4.1.1.2 Hardware machines and roadmaps

4.1.1.2.1 Scaling of the proposed algorithm

As it was described above, the potential speed-up for both QA and QAOA is believed to be bounded from above by quadratic scaling. The total solution time for these approaches is comprised of the time for a single run as well as the number of repetitions needed to get a satisfactory result (sample complexity). The former is given by the annealing time, which usually scales inversely with the quadratic minimum energy gap [Edward2000] during the annealing process. For QAOA a similar time scaling in terms of gate depth is plausible.

4.1.1.2.1.1 Cost of computing

In the context of QA and QAOA, the number of logical qubits comprises both the number of binary variables within the cost function and the number of auxiliary variables needed to account for constraints in the form of inequalities. For an average mission planning problem, with parameters such as the number of satellites in the constellation, the number of *Acquisition Requests* (AR), the number of *Data Take Opportunity* (DTO), and the number of *Download Opportunity* (DLO), as reported in Table 2 from [RD-2], the requirement for logical qubits falls within the range of 10^3 to 10^4 when considering preprocessing.

When implementing this problem on D-Wave hardware with the current Pegasus graph, the estimated number of physical qubits needed typically falls within the order of 10^4 to 10^5 qubits. It's worth noting that while the D-Wave Advantage 2 QPU currently provides access to 7000+ qubits, the D-Wave roadmap holds the promise of devices with improved connectivity, starting from 2025. In an optimistic scenario, these improved QPUs could potentially handle problems with up to 10^3 logical variables.

Furthermore, QPUs based on Rydberg atoms are also making significant strides in terms of hardware development. These systems are capable of implementing adiabatic protocols of a QUBO formulation, when operating in analog mode [Wurtz2022].

We roughly estimate the computing error and running time for the QAOA algorithm in terms of two-qubit gates in the NISQ regime. For this estimate we assume that the number of two-qubit gates scales as $n^2\log(n)$ where n is the number of qubits; this scaling can be expected to be a best-case scenario, or lower bound on the computing cost.

Assuming a two-qubit error of 7.4×10^{-3} for superconducting qubits, or 1×10^{-3} for trapped ion qubits, we plot the accumulated error that arises when running the QAOA algorithm on the left-hand side of Figure 1. To get a feeling for the expected runtime, we further plot the duration of such calculations on the right-hand side of Figure 1, based on two-qubit gate durations of 533 ns or 50 ms for superconducting and trapped ion qubits, respectively.

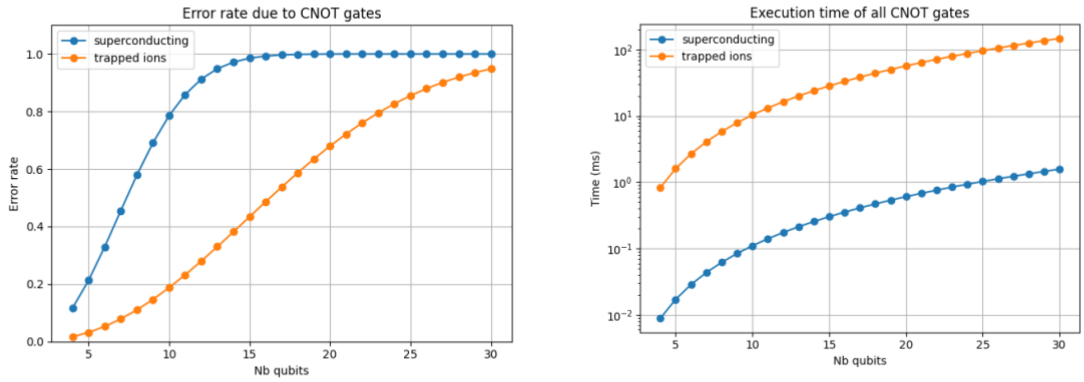


Figure 1: Error rates and execution time of QAOA: left: Approximation of accumulated error rates for QAOA, right: Approximation of runtime for QAOA.

From the plot of the error rate in Figure 1, we take that the error approaches 100% for less than 50 qubits. These current error rates become prohibitive even for small problem sizes, since measurement outcomes of such quantum circuits contain too little information.

There are various paths that can be followed from here. One is to consider significantly improved error rates. Another path is to consider the effect of error mitigation, which is a technique that alters the qubit operations with the aim of suppressing the accumulated error during the calculation to some degree, without correcting errors completely. This way the slope of the accumulated error, like that in Figure 1, becomes more gradual, thus allowing for larger parameter sets. Indeed, the IBM Quantum team has recently reached deep circuits with many quantum gates by employing a type of error mitigation [Kim2023].

For the problem at hand, however, we consider a third approach, which is that of quantum error correction. In this case, logical qubits are represented by several physical qubits, thereby introducing a redundancy that allows the correction of most errors.

As noted above, in an optimistic scenario the number of relevant quantum gates scales as $n^2\log(n)$ with the qubit number n. This scaling is roughly equal to that of Shor's famous factoring algorithm. For the factoring problem, [Wilhelm2020] has computed the resources necessary to factor a 1024-bit integer (which would be the task to break RSA-1024 encryption), which requires more than 1024 logical qubits. Their result is that roughly 10^{11} physical superconducting qubits are needed to carry out this calculation.

In the mission planning problem, problem instances whose solution requires between roughly 100 to 1000 logical qubits are instances that are hard to solve exactly. To arrive at a tangible number of qubits that are required to solve this problem, we use the comparison to Shor's algorithm and the calculation carried out in [Wilhelm2020] mentioned above. From this, we infer that an error corrected quantum computer based on superconducting circuits with 10^{11} physical qubits would likely suffice to solve the mission planning problem with 1000 logical qubits via QAOA in a reliable manner.

4.1.1.2.2 Expected time availability

Minimum size problem	Full size problem	Quantum maturity within 15 years	Timeline		
			Up to 5 years	Up to 10 years	Up to 15 years
[10^3 logical binaries example ~2 satellites, ~100 requests, ~10 DTO-DLO] Possible implementation: NISQ-compatible Ion-traps QC (low gate error rates), Superconducting qubits QC (fast gate operations); SC quantum annealer and neutral Rydberg atoms in optical lattice	[10^8 logical binaries example ~10^2 satellites, ~50K requests, ~10^2 DTO-DLO] Challenges: Large number of required qubits. Close to all-to-all connectivity	(High) Superconducting qubits QC QTRL 8-9 (High) Trapped ions QC QTRL 8 (High) Neutral Rydberg atoms in optical lattice QTRL 7-8 (Medium) Superconducting quantum annealer QTRL 5	Minimum size problem implemented on Superconducting quantum annealer and neutral Rydberg atoms in optical lattice	Minimum size gate-based problem implemented on ion-traps QC and superconducting qubits QC (without error correction)	Optimistic: Full size gate based problem implemented on NISQ superconducting qubits QC with very low error rates and fast gate times Pessimistic: Full size problem implemented on NISQ ion-trap QC with very low error rates and fast gate times Full size problem implemented on Rydberg atoms

We note that it does not seem impossible that a fully error corrected quantum computer with 10^{11} qubits will be realized within the next 15 years. For example, the quantum computing roadmap of Google Quantum AI leads up to the development of an error-corrected quantum computer with 1 million qubits,[1] which was announced to be realized by 2030.[2]

4.1.1.3 Bibliography

[Edward2000] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Michael Sipser: “Quantum Computation by Adiabatic Evolution”, 2000; arXiv:quant-ph/0001106].

[Fahri2014] Fahri, E. A quantum approximate optimization. arXiv:1411.4028, 2014.

[1] https://ai.google/static/documents/approach-quantum-computing.pdf

[2] https://qubitreport.com/quantum-computing-science-and-research/2021/05/20/1-million-qubits-by-1-1-2030-an-ambitious-goal-from-google-and-how-to-get-there/

[Stollenwerk2020]	T. Stollenwerk et al., "Quantum Annealing Applied to De-Conflicting Optimal Trajectories for Air Traffic Management," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 1, pp. 285-297, Jan. 2020, doi: 10.1109/TITS.2019.2891235.
[Zhou2020]	Zhou, L., Wang, S.T., Choi, S., Pichler, H., & Lukin, M. (2020). Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices. Phys. Rev. X, 10, 021067.
[Kim2023]	Y. Kim, A. Eddins, S. Anand, et al., Evidence for the Utility of Quantum Computing before Fault Tolerance, Nature 618, 500 (2023)
[Wilhelm2020]	F. K. Wilhelm, R. Steinwandt, B. Langenberg, et al., Status of quantum computer development, published via the German Federal Office for Information Security (2020), https://www.bsi.bund.de/qcstudie
[Wurtz2022]	Wurtz, J., Lopes, P., Gemelke, N., Keesling, A., & Wang, S. (2022). In Industry applications of neutral-atom quantum computing solving independent set problems.

4.1.2 Scenario 1b: Quantum machine learning

4.1.2.1 Proposed algorithms

4.1.2.1.1 Current performances of the selected algorithms

The algorithm considered to solve the mission-planning problem is a hybrid *Quantum Neural Network* (QNN) used as a policy model in a *Reinforcement Learning* (RL) environment. The quantum part of the algorithm is a quantum circuit added to the classical neural network as an ultimate layer. Currently, such an algorithm showed promising results in solving the mission-planning problem for small satellite constellations [Rainjonneau2023]. A 4-qubit parameterized quantum circuit has been simulated to solve a mission-planning problem with 2 satellites and 2000 requests, and has demonstrated a higher acquisition completion rate than a classical greedy algorithm.

Nowadays, classical algorithms with meta-heuristics are used to tackle the re-ordering of requests considering their level of priority for large constellations of satellites (10 to 100 satellites) and tens of thousands *Acquisition Requests* (AR). However, these methods do not guaranty a “good” solution for such large problems. Even if no quantum speedup has been demonstrated so far, expectations are that quantum algorithms may provide a better time-to-solution in the upcoming decades, or a “better” solution than current state-of-the-art in a similar computation time duration.

4.1.2.1.2 High-level description of algorithms implementation

Two approaches have been considered in the implementation of the QNN in the work proposed by TerraQuantum and Thales Alenia Space [Rainjonneau2023]. They consist in hybrid classical-quantum versions of the *Proximal Policy Optimization* (PPO) and AlphaZero algorithms. In both cases, the relevant parameters of the constellation and Acquisition Requests are used as features, and fed into a neural network. The first layers of the neural network are classical. They aim to reduce the number of relevant features which are eventually used as inputs to a parameterized quantum circuit. The outputs of the quantum circuit, i.e. the last layer of the QNN, enable the RL agent to assign the requests to each satellite. Typically, the quantum circuit is composed of an ansatz, which parameters are taken from the values obtained from the classical part of the QNN, and is repeated multiple times before measuring the output values.

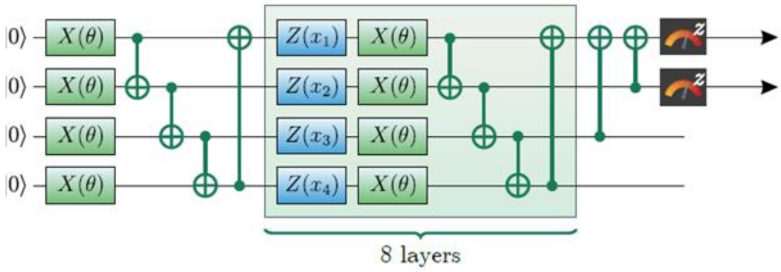


Figure 2. *Parameterized quantum circuit* proposed in [Rainjonneau2023]. The ansatz is highlighted in green.

4.1.2.1.3 Bottlenecks

The quantum circuit proposed in [Rainjonneau2023] is a 4-qubit circuit in which 32 features can be encoded due to the repetition of the ansatz over eight layers. A larger number of features could be encoded if either a larger number of layers are added, or a larger number of qubits is available, or both. However, this would significantly affect the performance of the algorithm, as the quantum part would be a very different circuit.

Estimating whether or not such modifications would lead to a better quality solution requires further investigations and a full study of the specific quantum circuits in this context. Nevertheless, the number of classical layers of the QNN could be reduced by increasing the number of encoded features in the quantum circuit. This could potentially lead to a faster execution time of the algorithm depending on the hardware performances. Still, a phenomenon occurring in machine learning approaches, known as the vanishing gradient problem, may occur. This phenomenon occurs in the backpropagation process in the training of artificial neural networks and can lead to suboptimal solutions. Such issue also occurs for large QNN and lead to the appearance of barren plateaus. According to [Kwak2021], the probability of occurring barren plateaus increases exponentially with the number of qubits. No clear solution has been found for large scale QNN yet, although the problem can be avoided for small scale QNN by setting good initial parameters.

Although the number of qubits is likely not a bottleneck for such algorithm, other hardware constraints pose serious challenges to their efficient implementation [Zaidenberg2021]. The limited depth of the quantum circuit that the hardware can handle while maintaining a near-optimal solution is to be taken into account if one modifies the quantum circuit to a larger one. The larger the quantum circuit is, the longer the qubit decoherence time must be in order to perform operations with a satisfying fidelity. Furthermore, the arrangement of qubits is also an important limitation that depends strongly on the entangling capabilities and thus the arrangement/number of entangling gates and SWAP gates in the quantum circuit. All these limitations come back to the problem of minimizing the error produced in the computation in the NISQ era.

4.1.2.1.3.1 Original kernel simplification for NISQ applications

Due to the small number of qubits required for the quantum circuit already studied, the algorithm can already be implemented in NISQ computers and do not require further simplifications. A small size problem has already been solved using quantum simulators. It is expected that such problems can already be solved by current NISQ hardware, especially superconducting and trapped ions platforms.

4.1.2.2 Hardware machines and roadmaps

The hardware requirements of this algorithm mostly consist in having a low enough error rate to faithfully execute the quantum circuit. The constraints in terms of number of qubits is met by most hardware technologies. The most promising technologies for small circuits such as the 4-qubit circuit studied are superconducting qubits and trapped ions.

4.1.2.2.1 Scaling of the proposed algorithm

The hybrid classical-quantum algorithm proposed makes use of a small quantum parameterized circuit and does not necessarily need to become larger to tackle full operational problems. It is the role of the classical part of the QNN to reduce the number of features down to the number of inputs used in the quantum circuit. Nevertheless, it might be beneficial to study larger/deeper quantum circuits in order to reduce the classical burden and potentially speed up the training phase.

In any case, the number of qubits is expected to remain reasonable. Quantum computers based on superconducting qubits already allow the implementation of circuits of hundreds of qubits when ion traps quantum computers currently face difficulties reaching more than 10 qubits. It appears that both technologies can be used to perform such small quantum circuits.

These technologies must be compared regarding the error rate obtained by executing this circuit and their potential achievements in the next years concerning gate fidelities, coherence times and connectivity. Both are very promising platforms, with a current QTRL estimated to 5-6 and successful demonstrations of quantum error correction experiments on small QPUs. Their QTRL is expected to be of 8-9 within the next 10 years, achieving scalable versions of their quantum computers and potentially exceeding the power of classical computers.

4.1.2.2.1.1 Cost of computing

The cost of computing depends on the fidelity of the gates, the gate times as well as the circuit depth. Ion traps quantum computers have complete connectivity while current superconducting qubits quantum computers have a relatively low connectivity in mainstream architectures. However, considering the small number of qubits used in the proposed circuit, this advantage is not significant here. Furthermore, superconducting qubits show promising developments in view of long-range connectivity for the future as well.

Both technologies have shown significant improvements in entangling gate fidelities over the years with values about 99.7% for superconductors and 99.9% for ion traps in 2020. When considering a small size circuit (in particular, 4 qubits and 8 layers), it appears that NISQ implementations are less subject to errors when using ion traps quantum computers. Ion traps computers would therefore be a good choice for near term implementations of small-scale problems.

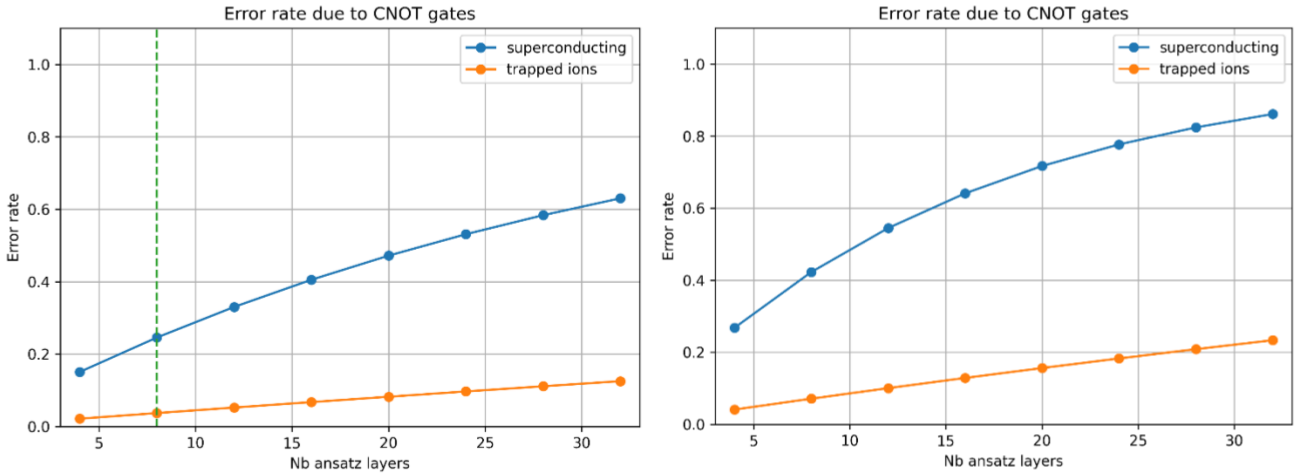


Figure 3. *Error rate produced by the CNOT gates in the quantum circuit studied, as a function of the number of ansatz layers. Left: 4-qubit circuit, Right: 8-qubit circuit. The green dotted line corresponds to the circuit presented in* **Figure 2**.

No restriction on the size of the problem to solve has been made. However, the larger the problem size, the more data need to be processed in the early (classical) steps of the neural network. Such a quantum circuit could thus be used for large scale problem as well but improvements in the error rates and gate times are needed in order to achieve relatively good quality solutions. However, a quantum speedup in the time-to-solution may not be significant since a large part of the computation is classical. Increasing the size of the circuit in order to perform a larger fraction of the computation in a quantum manner is a possibility but a trade-off is to be made between the size of the circuit used and the fidelity of its execution as shown in the previous graphs. In that regard, superconducting qubits quantum computers may very likely prove to be more interesting in the future. Indeed, their gate times are much faster (by two orders of magnitude) and gate fidelities will certainly be improved in the next years.

4.1.2.2.2 Expected time availability

Minimum size problem	Full size problem	Quantum maturity within 15 years	Timeline		
			Up to 5 years	Up to 10 years	Up to 15 years
2 satellites, 2.000 requests Possible implementation: NISQ-compatible Ion-traps QC (low gate error rates), Superconducting qubits QC (fast gate operations)	10-100 satellites, 10-100K requests Challenges: Large amount of data as inputs to the QNN, long training phase, very low error rate needed and fast gate times, possible barren plateaus for larger circuits	(High) Superconducting qubits QC QTRL 8-9	Minimum size problem implemented on NISQ devices (ion-traps, superconducting qubits)	Optimistic: Full size problem implemented on NISQ superconducting qubits QC with very low error rates and fast gate times Same algorithmic architecture (and larger quantum circuits if judged promising)	Pessimistic: Full size problem may require fully scalable error corrected quantum computers to be addressed

4.1.2.3 Bibliography

[Rainjonneau2023] Rainjonneau, Serge, et al. "Quantum algorithms applied to satellite mission planning for Earth observation." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* (2023).

[Kwak2021] Kwak, Yunseok, et al. "Quantum neural networks: Concepts, applications, and challenges." *2021 Twelfth International Conference on Ubiquitous and Future Networks (ICUFN)*. IEEE, 2021.

[Zaidenberg2021] Zaidenberg, Daniela A., et al. "Advantages and bottlenecks of quantum machine learning for remote sensing." *2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS*. IEEE, 2021.

4.2 Use case n°2: Multiple-view Geometry on Optical Images

4.2.1 Proposed algorithms

4.2.1.1 Current performances of the selected algorithms

The feature-extraction and matching problem between multiple satellite optical images requires heuristics, problem decomposition and a large computational power to be solved classically. The quantum approach considered in this study consists in extracting relevant keypoints of the images in a first part, and matching the keypoints of different images in a second part. The study conducted by Piatkowski et al. [Piatkowski2022] showcases that quantum algorithms could already perform these tasks for small images of a few tens of pixels using a quantum annealing approach. Both the keypoint extraction problem and the feature-matching problem are solved using D-Wave's quantum annealer. No quantum speedup has been observed yet but future developments of the technology might provide a more efficient way to solve these problems than classical approaches.

4.2.1.2 High-level description of algorithms implementation

Two quantum algorithms are offered in order to extract keypoints from a given image. Both these algorithms rely on efficient clustering methods. The first one, quantum k-medoids, consists in minimizing the distance between the keypoints and the rest of the pixels while maximizing the distance between two distinct keypoints. This optimization problem is solved on a quantum annealing machine using a QUBO formulation. The second one, quantum kernel density clustering, consists in finding cluster centroids such that two feature map distributions are as similar as possible. This optimization problem is also solved using a QUBO formulation. However, the kernel matrix that defines the feature map is computed via a 4-qubit quantum circuit. This circuit transforms a uniform superposition via n-qubit unitary operations. Here, the data of two features *x* and *y* is passed as parameters of these universal unitary gates. The choice of the operators is not fixed but it is expected that a potential quantum speedup could only be possible if the feature map cannot be efficiently simulated classically.

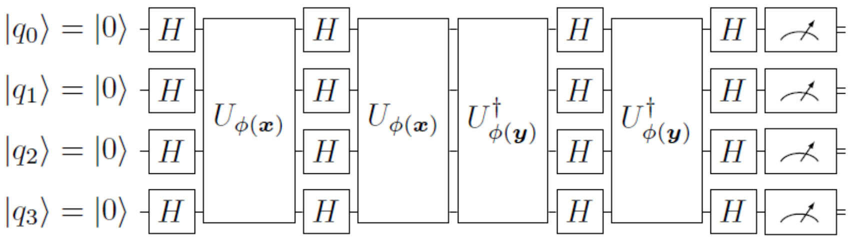


Figure 4. *Quantum circuit to construct the kernel matrix [4].*

The feature matching part is also conducted on a quantum annealer. It consists in identifying matches between keypoints of two distinct images using feature descriptors such as *Scale Invariant Feature Transform* (SIFT). The optimization problem is also formulated as a QUBO and the kernel function previously mentioned can be used to access a high-dimensional feature space.

4.2.1.3 Bottlenecks

Satellite optical images of interest are composed of 3000x3000 pixels up to 30000x30000 pixels on 3 RGB channels or 4 channels. Simply encoding such large data using a quantum annealing approach described above would require up to tens of millions of qubits, which cannot be done without *Fault-Tolerant Quantum Computing* (FTQC). This presents a first major bottleneck but it is possible to circumvent it by dividing the keypoint extraction task of a full image in the same task to smaller parts of the initial image.

In addition, even though the number of qubits for the circuit realizing the feature map does not, the circuit depth increases significantly with the number of pixels. The challenge is to find an efficient implementation of the n-qubit operators used in the quantum circuit. Here, the suggested operations are the ones proposed in [Piatkowski2022, Havlíček2019] where N is the number of pixels.

$$U_{\phi(\boldsymbol{x})} = \exp\left(-i \sum_{S \subset [N]} \phi_s(\boldsymbol{x}) \prod_{v \in S} \sigma_z^v\right)$$

This implies that the quantum circuit is composed of a large number of CNOT gates and multi-qubit gates, which would later be transpiled into a set of SWAP gates and two-qubit gates. This design can also be simplified as it has been done in [Piatkowski2022].

Having a larger number of qubits in this circuit does not necessarily help achieving a quantum advantage because all the qubits end up being measured. Moreover, it is necessary to run this circuit a number of times that is quadratic with the number of data points in order to determine the full kernel matrix. However, the measurement operation is one that destroys any potential quantum advantage when applied excessively.

The feature matching process faces the same problem, which is the limited number of qubits allowed by the hardware machine. The larger the number of qubits available, the larger the images this algorithm can process.

4.2.1.3.1 Original kernel simplification for NISQ applications

The process of keypoints extraction from a very large image can be subdivided into the same process for much smaller images in order to reduce the hardware constraints concerning the number of qubits needed to perform the computation. The idea is to extract a small number of keypoints in all the small images (of hundreds or thousands of pixels) and iteratively extract some of these keypoints by considering aggregates of such sub-images until the desired number of keypoints have been extracted from the initial image. This bypasses the issue of accessing a large number of qubits in this algorithm but remains the need for high connectivity between qubits to limit the number of operations and thus the error that would result from them. This intermediate step consists in solving a medium size problem rather than directly extracting keypoints from the full initial image. This medium size problem could potentially be solved in the NISQ era.

Concerning the quantum circuit for the computation of the kernel matrix, the operation considers local feature functions for all subset of $S = [N]$, which involves a series of multi-qubit gates that drastically increase the depth of the circuit when transpiled into the native gates of the hardware. This approach is too costly and it is recommended to consider only pairwise features (which involve two-qubit gates), i.e. $|S| = 2$, for the algorithm to be NISQ-compatible.

Such simplifications are legitimate and do not hinder the resolution of the initial problem, but only future hardware developments will allow to proceed without them.

4.2.2 Hardware machines and roadmaps

The scaling of the proposed algorithm, and the size of the problems that can be solved, depends on the hardware capabilities to provide an all-to-all connectivity, a large number of qubits and to execute gate operations in a short time and high fidelity.

4.2.2.1 Scaling of the proposed algorithm

On the one hand, the QUBO instance of the keypoint extraction algorithm can currently be handled by superconducting quantum annealers as long as the images are reduced to the size of a few tens of pixels. Even though the number of qubits in such devices may be much larger than necessary, their performance is limited by the connectivity between these qubits. On the other hand, cold atoms quantum computers have native all-to-all connectivity but scaling up beyond thousands of qubits is a serious technological challenge.

The quantum kernel clustering approach makes use of a quantum circuit with a small number of qubits. However, this circuit is to be run a quadratic number of times compared to the number of data points. Emphases should be made on the necessity of having access to high gate fidelities. The circuit depth is also relatively small when considering the simplified version of the operators, which makes ion traps and cold atoms promising hardware technologies considering their native all-to-all connectivity as well. It is possible to consider a quantum circuit with a larger number of qubits to access a higher dimensional feature space but the circuit depth and the number of 2-qubit gates would increase significantly as they depend on the number of combination of all qubits taken two at a time C_2^n. As mentioned in the previous section, increasing the number of qubits implies increasing the number of measurements, which can potentially hinder any potential quantum speedup. The circuit described in this study can be executed on NISQ hardware based on cold atoms, ion traps or superconducting qubits with a specific design to meet the connectivity requirements.

The feature-matching problem uses the quantum circuit that represents the feature map as well as a QUBO instance to solve the corresponding optimization problem. This algorithm faces the same challenges as the ones just discussed.

In order to efficiently solve the multiple view geometry problem on a full size optical image, the quantum annealer utilized must be able to efficiently use millions of qubits. This would therefore require a fully scalable error-corrected quantum computer, which will likely not be available in the next decade. In addition, the quantum circuit would have to be executed billions of times in order to compute the whole kernel matrix. The error rate of gate operations would need to be extremely low in order to maintain a satisfying level of fidelity of the results. Whether this would lead to a better quality solution than what is achievable classically is still an open question.

4.2.2.1.1 Cost of computing

The cost of computing for one execution of the quantum circuit is estimated from its circuit depth, the fidelity of the 2-qubit gates and their gate times. The following estimates of computing resources are those achieved when running the series of CNOT gates contained in the circuit described above, depending on the number of qubits, with the current state of superconducting technology and ion traps hardware.

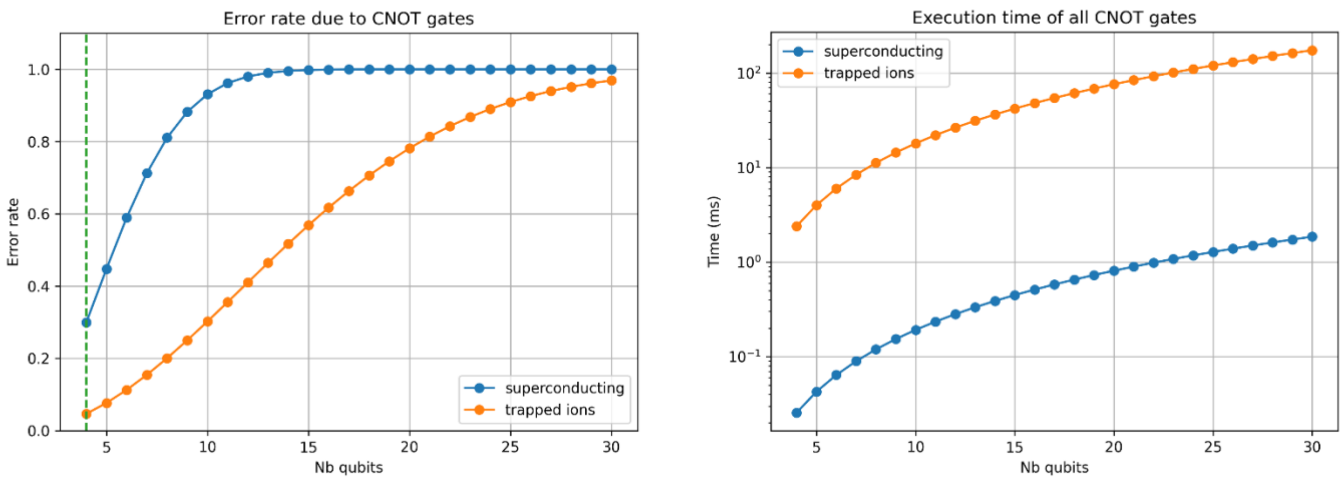


Figure 5. **(a)** *Error rate produced by the CNOT gates in the quantum circuit studied, as a function of the number of the number of qubits. The green dotted line represents the circuit presented in* **Figure 4**. **(b)** *Time to execute the ensemble of all 2-qubit gates in the circuit.*

Even though both technologies can be used as of now to execute the proposed algorithm, their advantages differ. The ion traps based machines allow for a better fidelity when implementing a 2-qubit gate but it come at the cost of a very slow execution of the circuit when compared to superconducting qubits. This is particularly noticeable as the size of the circuit considered is larger. Typically, a superconducting machine would be faster by two orders of magnitude (0.025ms versus 2.5ms for a single execution of the 4-qubit circuit presented above). These estimates are computed to highlight only the trend of execution time and their order of magnitude. They do not stand as real execution time values. The CNOT here are assumed to be executed one after the other but some of them can be executed in parallel. The real execution time would therefore be lower than those indicated in the graphs.

Furthermore, the 2-qubit gate error rates may seem relatively small for the ion-traps technology but the circuit is run a large number of times within the algorithm, $N(N-1)/2$ times where N is the number of pixels of the image. It is therefore necessary to improve the gates fidelities in view of solving the keypoint extraction and feature matching problems on large images.

In this perspective, ion traps may be viewed as the most promising technology in the long term due to their larger gate fidelities and all-to-all connectivity, if the circuit chosen remains small. Superconducting qubits may also be promising if a large connectivity can be achieved with some clever architecture and if the gate fidelities are improved.

4.2.2.2 Expected time availability

Minimum size problem	Full size problem	Quantum maturity within 15 years	Timeline		
			Up to 5 years	Up to 10 years	Up to 15 years
Extraction of 10 keypoints on 8x8 patches Feature matching of 10 keypoints <u>Possible implementation:</u> NISQ-compatible Superconducting Q annealer for QUBO Ion-traps/cold atoms (low gate error rates and all-to-all connectivity), Superconducting qubits QC (fast gate operations)	3099x2029 pixels up to 30Kx30K pixels <u>Challenges:</u> Large amount of data thus large number of qubits required (millions), very low error rates and fast gate times needed for the numerous runs of the circuit	(High) Superconducting qubits QC QTRL 8-9, Ion traps QC QTRL 8	Minimum size problem implemented on NISQ devices (ion-traps, superconducting qubits) <u>Optimistic:</u> Medium size problem (~1000s pixels) implemented on NISQ devices	<u>Pessimistic :</u> Medium size problem (~1000s pixels) implemented on NISQ devices (large number of qubits for QA with high connectivity, low error rates and gate times for quantum circuit)	Full size problem requires fully scalable error corrected quantum computers to be solved

4.2.3 Bibliography

[Piatkowski2022] Piatkowski, Nico, et al. "Towards Bundle Adjustment for Satellite Imaging via Quantum Machine Learning." *2022 25th International Conference on Information Fusion (FUSION)*. IEEE, 2022.

[Havlíček2019] Havlíček, Vojtěch, et al. "Supervised learning with quantum-enhanced feature spaces." *Nature* 567.7747 (2019): 209-212.

4.3 Use case n°3: Optical Satellite Data Analysis

4.3.1 Proposed algorithms

4.3.1.1 Current performances of the selected algorithms

In [Havlicek2018], one of the first work dealing with application of quantum kernels to *Machine Learning* (ML), a *Support Vector Machine* (SVM) employing a quantum-generated kernel managed to classify an artificially generated binary classification dataset with 100% accuracy. However, it is important to point out that the dataset was specifically designed so that it could be efficiently classified by the quantum kernel and that the training and test size were small, consisting of 20 samples each. In the study it was also shown that there are some quantum kernels that are conjectured to be hard to simulate classically thus setting the ground for potential quantum advantage in case such quantum kernels were shown to provide better prediction results. However, how and whether quantum kernels methods could provide advantage in terms of accuracy over classical ones is still an open area of research.

More recently, in 2023, a study employing quantum kernels with *Support Vector Machines* to a cloud detection problem was conducted [Miroszewski2023]. In such a work different quantum kernel implementations were tested with datasets of different training sizes and the results compared with SVMs employing classical well-known kernels. As the training size increased (values ranging from 10 to 1280 were considered) the accuracy of some quantum kernel implementation roughly matched the one achieved by the classical *Radial Basis Function* (RBF) kernel. The accuracy achieved, however, was still slightly lower than the accuracy achieved by current state of the art deep learning methods employing neural networks with convolutional layers.

4.3.1.2 High-level description of algorithms implementation

The aim of *Quantum Kernel Estimation* is to use *Parametrized Quantum Circuits* (PQC) to implement a kernel function that is then used with ML kernel classification models. In this setting the PQC, which in this case is also referred to as “feature map”, is used to encode the feature vectors in corresponding quantum states. The parameters of the feature map circuit are dependent on the feature vector entries and is therefore possible to associate to an input feature vector the corresponding encoded quantum state. The quantum kernel function value between two feature vectors from the dataset is then obtained by considering the fidelity between the encoded quantum states. The fidelity between quantum states cannot be accessed directly so it is necessary to estimate it through a sampling procedure.

4.3.1.3 Bottlenecks

One of the main bottlenecks of classical kernel methods is that they require a number of evaluations that scale quadratically with the training size. This characteristic is also inherited by the quantum kernel and thus potential quantum speed-ups are to be searched for in how the usage of a quantum kernel might provide advantage over a classical one in terms of performances rather than a speed-up in computational complexity.

The number of qubits needed for the quantum kernel algorithm is strictly related to the number of features of the datapoints. When considering optical remote sensing images, the number of spectral bands is about a few tens whereas for some hyperspectral images it might reach a few hundred bands. Usually, the encoding of the feature map is structured such that the number of qubits equals the number of features of the data points.

To efficiently implement a feature map that cannot be easily simulated classically many 2-local qubit gates are usually needed, thus requiring a high qubit connectivity that is not available in current superconducting architectures [Russo2023].

Due to the sampling error that arises when estimating the fidelity through sampling, the Gram matrix, i.e. the matrix storing the kernel function evaluations between all data points, might not be positive semi-definite as required by kernel theory. Therefore, some post-processing is needed to regularize the obtained matrix into a positive-definite one [Hubregtsen2021].

4.3.1.3.1 Original kernel simplification for NISQ applications

The number of qubits required in a circuit to implement a quantum kernel depends on several factors. Among the main ones are the number of features of the training samples, the encoding procedure defined by the feature map and the strategy used to estimate the fidelity between quantum states that is used as kernel function evaluation. In some cases, therefore, it might be useful to perform feature reduction in order to reduce the number of required qubits.

4.3.2 Hardware machines and roadmaps

4.3.2.1 Scaling of the proposed algorithm

The required number gate operations to implement a quantum kernel depends on the circuit structure of the feature map. Moreover, to achieve entanglement in the encoded state, the number of two-qubits gate operations scales with the number of qubits. The order of the scaling depends on the feature map: for example, the implementation of the ZZ feature map by IBM Qiskit allows to choose different encoding strategies having linear or polynomial scaling in the number of gates to achieve entanglement in the encoded state. The estimation of the fidelity between quantum states also affects the usage of auxiliary computing resources: the estimation through the adjoint circuit approach does not require extra qubits at the expense of doubling the circuit depth, whereas the SWAP test does not affect the depth of the circuits but requires a total of 2n+1 qubits, with n being the number of qubits in the feature map [Hubregtsen2021]. Moreover, since the fidelities between quantum states are obtained through a sampling procedure, the number of samples needed will depend on the precision that one wants to achieve. Finally, as mentioned earlier, the quantum kernel inherits the quadratic scaling in the number of kernel evaluations from classical kernel methods.

4.3.2.1.1 Cost of computing

It is not possible to estimate, in general, the cost of computing for every quantum feature map, as it depends on the circuit structure of the map itself. We therefore chose to estimate the cost of computing for the ZZ feature map as defined in the Qiskit software library as an example. We chose a structure of the feature map with one repetition, meaning that the circuit is only repeated once, and a full entanglement structure which entails a quadratic scaling of the number of CNOT gates needed.

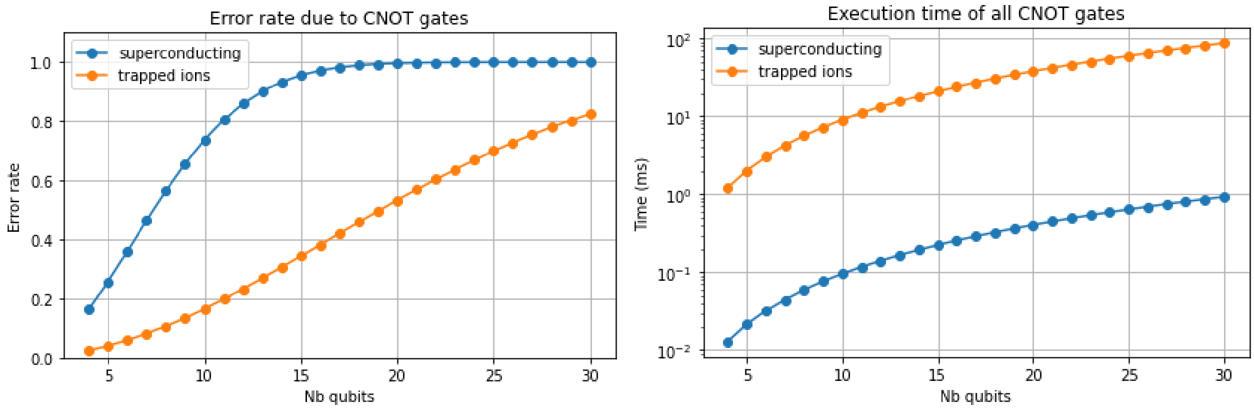


Figure 6. (a) *Error rate produced by the CNOT gates in the ZZ feature map with one repetition and full entanglement strategy, as a function of the number of the number of qubits.* **(b)** *Time to execute the ensemble of all 2-qubit gates in the circuit as a function of the number of qubits*

In **Figure 6** the error rate and the execution time are shown as a function of the number of qubits for the considered feature map. The total number of CNOT operations required for this specific feature map architecture scales quadratically with the number of qubits.

The number of repetitions of the feature map circuit does not affect the nature of the scaling since it will correspond to a scalar factor in the computational cost function.

Regarding the hardware implementation, since for quantum kernels the number of qubits required is not particularly high, the ion trap hardware might be a good choice since it currently provides better error rates than superconducting qubits, however, the ion trap hardware also entails a higher execution time.

4.3.2.2 Expected time availability

Minimum size problem	Full size problem	Quantum maturity within 15 years	Timeline		
			Up to 5 years	Up to 10 years	Up to 15 years
Learning problem with 1000 training samples. The nature of the data points affects the size of the image being investigated <u>Possible implementation:</u> NISQ-compatible Superconducting Ion-traps/cold	Typical image size has few thousand pixels per row and column <u>Challenges:</u> Large number of quantum state fidelities that need to be estimated (scales quadratically with dataset size)	(High) Superconducting qubits QC QTRL 8-9, Ion traps QC QTRL 8	<u>Optimistic:</u> Larger problems with several thousand qubits might be solved with real hardware	Medium size problem implemented on NISQ devices	Full size problem might be possible, but the compute time might still be very high due to the high number of fidelities to be estimated.

atoms (low gate error rates and all-to-all connectivity), Superconducting qubits QC (fast gate operations)					

4.3.3 Bibliography

[Havlicek2018] HAVLÍČEK, Vojtěch, et al. Supervised learning with quantum-enhanced feature spaces. *Nature*, 2019, 567.7747: 209-212.

[Miroszewski2023] MIROSZEWSKI, Artur, et al. Detecting Clouds in Multispectral Satellite Images Using Quantum-Kernel Support Vector Machines. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 2023.

[Russo2023] RUSSO, Marco; GIUSTO, Edoardo; MONTRUCCHIO, Bartolomeo. Quantum Kernel Estimation With Neutral Atoms For Supervised Classification: A Gate-Based Approach. *arXiv preprint arXiv:2307.15840*, 2023.

[Hubregtsen2021] HUBREGTSEN, Thomas, et al. Training quantum embedding kernels on near-term quantum computers. *Physical Review A*, 2022, 106.4: 042431.

4.4 Use case n°4: SAR Raw Data Processing

4.4.1 Proposed algorithms

4.4.1.1 Current performances of the selected algorithms

Quantum Fourier Transform (QFT) has been a well-known quantum algorithm for decades. Small implementations of the QFT have been validated on real or simulated quantum devices. Acceptable levels of fidelity can be reached with a 6-qubit circuit, considering realistic estimates of current hardware noise [Martin2020]. Nevertheless, a direct advantage cannot be reached with respect to the classical Fourier Transform when used independently as part of an algorithm operating on classical data. For this reason, a full-quantum *Range Doppler Algorithm* (RDA) for the selected use case is proposed. The algorithm is still defined at a high level and no experiments have been performed. Nevertheless, its feasibility is strictly dependent on the feasibility of QFT.

4.4.1.2 High-level description of algorithms implementation

A definition of the quantum RDA is shown in **Figure 7**.

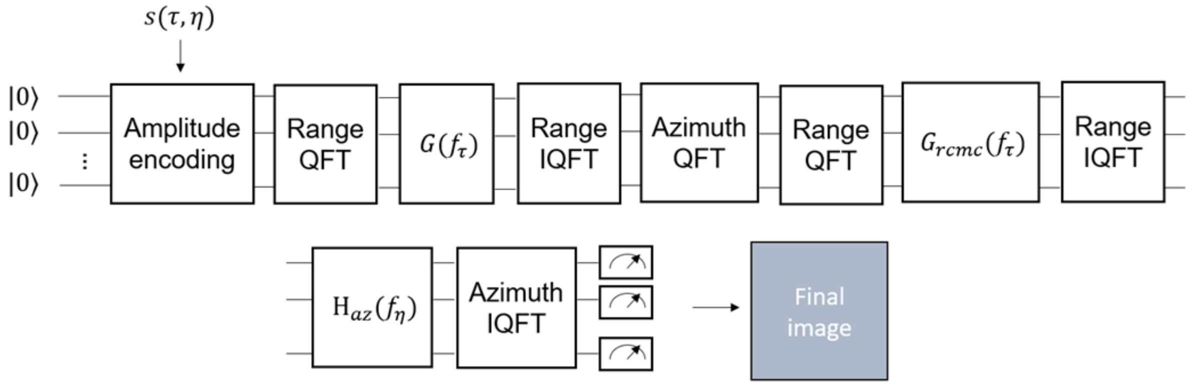


Figure 7. High-level scheme of the proposed quantum Range Doppler Algorithm (RDA).

The total number of pixels is $N_\tau \times N_\eta$. Encoding the complex values as quantum state amplitudes requires $n_\tau + n_\eta$ qubits, with $n_\tau = \log_2(N_\tau)$ and $n_\eta = \log_2(N_\eta)$. A multidimensional QFT can be separated into multiple QFT circuits applied to specific dimensions (range, azimuth) [Pfeffer2023]. The remaining correction operations required in the RDA are encoded as quantum circuits applied to the amplitudes. Their precise and efficient definition is not analyzed here, as it is expected to have a lower impact on the feasibility of the circuit compared to amplitude encoding and QFT.

4.4.1.3 Bottlenecks

The number of qubits required is logarithmic with respect to the size of the image. This can affect the scalability of the algorithm for large signal acquisitions when the number of available qubits is limited. However, the main bottleneck is the circuit depth, as a result of stacking multiple processing steps. It is known that the number of CNOT gates required to implement a QFT are $\Theta(N^2)$ [Plesch2011], whereas an amplitude encoding circuit requires $\Theta(2^N)$ CNOT gates [Nielsen2010]. The need for such a deep circuit goes far beyond the algorithmic trend in the NISQ era that considers small variational circuits.

4.4.1.3.1 Original kernel simplification for NISQ applications

Smaller areas can be analyzed, which then decrease the number of qubits needed to encode the total number of pixel values, and in turn reduce the circuit depth. A full-size image could be divided into multiple tiles separately processed. However, this affects the quality of the correction steps and the obtained image. The circuit cannot be further simplified from an algorithmic perspective, as the Range Doppler Algorithm contains the basic steps required to apply corrections to the acquired signal. Some circuit design methods can help develop quantum circuits with a lower depth at the cost of more ancillary qubits.

4.4.2 Hardware machines and roadmaps

4.4.2.1 Scaling of the proposed algorithm

The scaling depends on the chosen architecture. Superconducting qubits and ion traps are considered.

For the full processing pipeline to be executed, a sufficiently reliable hardware implementation, in terms of coherence time and gate fidelity, is ideally required. However, in the considered setting, this cannot be achieved in the near future. For this reason, scalable *Quantum Error Correction* (QEC) is a more realistic requirement. This mechanism can achieve reliable performance on machines with a large number of qubits, suggesting that superconducting qubits could be the main choice. The number of qubits is currently not a limiting factor, as the 8 qubits requirement for the minimum problem can be already achieved by both technologies.

4.4.2.1.1 Cost of computing

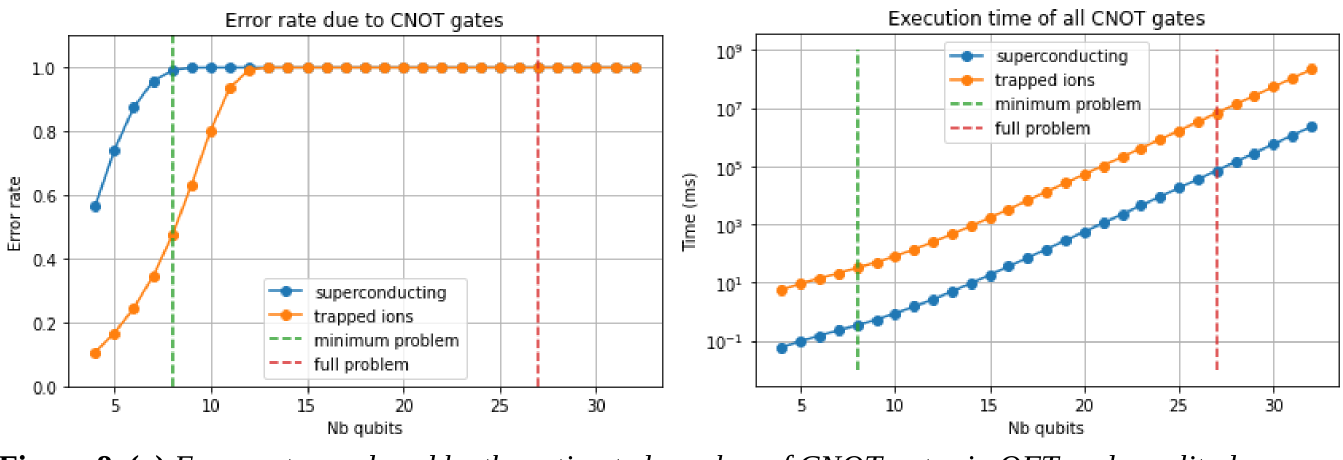


Figure 8. (a) *Error rate produced by the estimated number of CNOT gates in QFT and amplitude encoding, as a function of the number of qubits, for different computing technologies* **(b)** *Time to execute the ensemble of all 2-qubit gates in the circuit as a function of the number of qubits, for different computing technologies. Highlighted are the estimated minimum and full-size problem values.*

An estimate of the error rate and the execution time of the proposed circuit is shown in **Figure 8**. The performance of the different computing technologies takes into account the current SotA. These estimates are computed to highlight only the trend of execution time and their order of magnitude. They do not stand as real execution time values. The CNOT here are assumed to be executed one after the other, but the proposed circuit is inherently parallel. The contribution of the intermediate algorithmic blocks for correction is not included. The time overhead of a QEC algorithm cannot be estimated.

From the estimates, it can be seen that QEC is crucial for the execution of the algorithm. Despite the better error rates for trapped ions, the technology is still not mature enough. However, the higher QTRL expected in 10 years could bring trapped ions into consideration. The total execution time is acceptable. Note that the circuit does not require variational training or a large number of multiple executions.

4.4.2.2 Expected time availability

Minimum size problem	**Full size problem**	**Quantum maturity within 15 years**	**Timeline**		
			Up to 5 years	**Up to 10 years**	**Up to 15 years**
Image formation of a 16x16 patch (specific object and location) Possible implementation: Ion-traps/cold atoms (low gate error rates and all-to-all connectivity), Superconducting qubits QC (fast gate operations)	Image formation of a 10000x10000 patch (Sentinel-1 acquisition) Challenges: Large amount of data thus large number of qubits required, very low error rates and fast gate times needed for the deep circuit	(High) Superconducting qubits QC QTRL 8-9, Ion traps QC QTRL 8	No feasible implementation envisioned	Optimistic: Minimum size problem implemented on NISQ devices (ion-traps)	Pessimistic: Minimum size problem implemented on NISQ devices (ion-traps) Optimistic: Full size problem solved on fully scalable error-corrected quantum computers (superconducting)

4.4.3 Bibliography

[Martin2020] Martin, A., Lamata, L., Solano, E., & Sanz, M. (2020). Digital-analog quantum algorithm for the quantum Fourier transform. *Physical Review Research*, *2*. https://doi.org/10.1103/PhysRevResearch.2.013012

[Pfeffer2023] Pfeffer, P. Multidimensional Quantum Fourier Transformation. *ArXiv*, 2023.

[Plesch2011] Plesch, M., & Brukner, Č. Quantum-state preparation with universal gate decompositions. Physical Review A, 83(3), 032302, 2011.

[Nielsen2010] Nielsen, M. A., & Chuang, I. L. Quantum computation and quantum information. Cambridge university press, 2020.

5 Conclusion

5.1.1 Use case 1: Mission Planning for EO Acquisitions

In **scenario 1a**, we have proposed a formulation of the mission planning problem as an optimization problem, for which attainable solutions exist on both a general-purpose quantum computer utilizing the *Quantum Approximate Optimization Algorithm* (QAOA) and a *Quantum Annealer* (QA), or an analog quantum simulator. The primary challenge inherent to this problem stems from the **extensive qubit resources** it demands.

Although photonic platforms could potentially accommodate a greater number of qubits, our focus is primarily on forthcoming platforms such as **superconducting qubits** (quantum computers and quantum annealers), **ion-trap** quantum computers, and **Rydberg atoms** confined within an optical lattice.

We anticipate that the **minimum-size problem** can be addressed within the **next 3 to 5 years** using an analog approach, while the gate-based formulation will necessitate **5 to 10 years** due to the substantial **number of qubits** and **gates** required. However, the timeline for solving the **full-size problem** extends **beyond 15 years** for both approaches.

In **scenario 1b**, we have focused on a **hybrid classical-quantum** method based on a *Quantum Neural Network* (QNN) to address the mission planning problem. Promising results have already been obtained from the simulation of a small-scale instance of the problem in previous works. The main advantage of this approach is the **limited number of qubits** required for its quantum part, no matter the size of the problem considered.

Therefore, we expect that emerging quantum hardware technologies such as **ion-trap** quantum computers and **superconducting qubits** may be used to solve the **full-size problem** within the **upcoming decade** if their developments allow for the realization of very **low error rates** and **fast gate time**. Even though no quantum advantage has been demonstrated theoretically, further studies on the design of an optimal QNN could be an interesting direction to follow in view of obtaining a **quantum speedup** and/or a **better quality solution** to the problem.

5.1.2 Use case 2: Multiple-view Geometry on Optical Images

We have addressed the keypoint extraction and the feature matching problem using an optimization formulation and by utilizing both **gate-based quantum computers** and **quantum annealers**.

We envision that **ion-trap** quantum computers and **superconducting** quantum computers and **annealers** are the most promising platforms for this use-case considering the **needs for high qubit connectivity** and for a **large number of qubits**.

The small size problem, which addresses images of tens of pixels, can **already be solved** efficiently using this quantum approach even though no clear advantage is demonstrated. However, extracting keypoints directly from the original image of a full-size problem would require a **substantial amount of qubits** and **high connectivity** from the hardware architecture chosen. Such achievements seem unrealistic without **scalable error corrected quantum computers** and thus extend **beyond 15 years**. Nevertheless, an iterative approach that consists in solving the keypoint extraction problem on **batches of medium-size images** seems very promising. While the solution quality might slightly differ from that of the original approach, this process reduces greatly the resources needed for practical implementation on real hardware and may be possible within **5 to 10 years**.

5.1.3 Use case 3: Optical Satellite Data Analysis

We have proposed a study on the potential application of **quantum kernel methods**, which are implemented by encoding an input feature vector into a corresponding quantum state. The algorithm can be implemented on various quantum hardware platforms such as **superconducting** and **trapped ion** hardware.

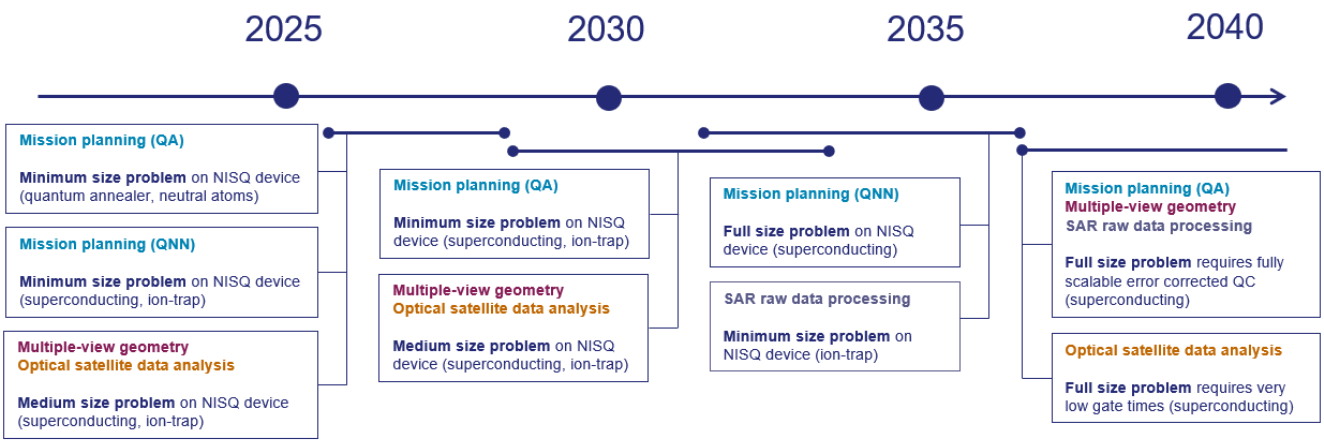
The number of qubits needed for this approach is strictly related to the number of features of data points and is not particularly high, requiring a **few hundred qubits** for hyperspectral data with many acquisitions at different spectral bands. The main obstacle for this approach is the **computational time** that is needed to calculate the kernel function evaluations between the data points. We anticipate that a **full-size problem** might be possible to solve **within the 15 years**' time frame. However, it is not yet clear whether the quantum kernel implementation might provide an advantage compared to classical solutions.

5.1.4 Use-case 4: SAR Raw Data Processing

We have proposed a quantum circuit version for the *Range Doppler Algorithm*, based on the *Quantum Fourier Transform*. SAR imaging is a highly relevant data-intensive approach in the context of Earth Observation, due to its effectiveness with different weather conditions. An algorithmic speedup can potentially be achieved when the whole processing pipeline is performed in the quantum domain. The **large circuit depth** poses a challenge for NISQ devices, as they would require relatively **low gate error rates** and **long coherence times**. **Ion-trap** devices may be able to solve a **minimum size problem** in the future, but **scalable** *Quantum Error Correction* can realistically be achieved only by **superconducting** devices. Optimistic forecasts envision this achievement **within the next 15 years**, also due to the low number of logical qubits required. Additional studies on the feasibility of the approach and its specific circuit implementation are required.

5.2 Graphical timeline

Comparing the industrial roadmaps of quantum computing platforms with the expected size of these use cases we can estimate an hypothetical timeline for the use cases:



5.3 Few directions and next steps (for the next 3 years) to start working

In order to demonstrate the advantage of a quantum approach on Earth Observation applications, further studies have been identified as interesting directions to start working, among which:

- Studies on the design of an optimal QNN in view of obtaining a **quantum speedup** and/or a **better quality solution** to the mission planning problem,
- Iterative approach that consists in solving the keypoint extraction problem on **batches of medium-size images,**
- Studies to analyze whether the **quantum kernel implementation** might provide an advantage compared to classical solutions for optical satellite data image classification,
- Additional studies on the feasibility of the **Quantum Fourier Transform** approach and its specific circuit implementation.

END OF DOCUMENT