



Dissertation

an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig-Maximilians-Universität München

Quantum Machine Learning for Large-Scale Classical Datasets with Applications in Earth Observation

eingereicht von

Soronzonbold Otgonbaatar

am 25. October 2024



Dissertation

an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig-Maximilians-Universität München

Quantum Machine Learning for Large-Scale Classical Datasets with Applications in Earth Observation

eingereicht von

Soronzonbold Otgonbaatar

am 25. October 2024

Erster Gutachter: Prof. Dieter Kranzlmüller, Ludwig-Maximilians-Universität München, Germany

Zweiter Gutachter: Prof. Mihai Datcu, POLITEHNICA București, Romania

Dritter Gutachter: Prof. Martin Werner, Technische Universität München, Germany

Tag der mündlichen Prüfung: 13. Februar 2025

Eidesstattliche Versicherung

(Siehe Promotionsordnung vom 12.07.11, § 8, Abs. 2 Pkt. .5.)

Hiermit erkläre ich an Eidesstatt, dass die Dissertation von mir selbstständig, ohne unerlaubte Beihilfe angefertigt ist.

Otgonbaatar, Soronzonbold

.....
Name, Vorname

München, 25. October 2024

.....
(Ort, Datum)

Soronzonbold

.....
(Unterschrift Doktorand/in)

Abstract

Big satellite datasets are used to monitor the Earth’s surface or for Earth observation challenges like land cover changes. They enable leveraging artificial intelligence, including machine and deep learning, for detecting changes on the ground. Compared to conventional computational algorithms, artificial intelligence helps find better optimal solutions to Earth observation challenges involving big datasets. Decision-makers and policymakers already extensively use these solutions to make fast, safety-critical, and human-centered decisions.

Quantum machine learning inspired by quantum algorithms promises to process some data-driven tasks faster than its conventional counterparts. We aim to use various benchmark satellite datasets to develop and benchmark quantum machine learning approaches with traditional artificial intelligence models. There are three main issues for processing quantum machine learning on benchmark satellite datasets: 1. which machine learning task for big satellite datasets and which satellite data-driven task can be efficiently and effectively processed on a quantum computer? 2. how to embed large-dimensional satellite data points in input quantum states, and 3. how to profit from both supercomputers and quantum computers.

To find a scientific answer to these three questions, we examine and identify both machine learning and satellite data-driven tasks that can be deployed on a quantum computer, otherwise inherently intractable. We then propose the encoding strategy of classical problems involving big satellite datasets in a quantum computer, named two-level encoding. Further, we design and investigate quantum machine learning approaches for a quantum annealer and a noisy intermediate-scale quantum computer for supervised learning tasks.

For supervised learning tasks, the performance of our quantum machine learning approaches is already competitive (and even better in some instances) compared to the ones of their classical counterparts. Additionally, we estimate the quantum resource required to gain an advantage over a supercomputer and profit from a supercomputer and a quantum computer. Doing so gives us insights into a future fault-tolerant quantum computer for tackling practical computational problems.

Kurzfassung

Man benutzt große Datensätze, um die Oberfläche der Erde zu überwachen oder für Erdbeobachtungsaufgaben wie Veränderungen der Bodenbedeckung. Diese Datensätze erlauben es, künstliche Intelligenz einschließlich maschinellem Lernen und Deep Learning einzusetzen, um Veränderungen auf der Erdoberfläche zu entdecken. Im Vergleich zu konventionellen Rechenverfahren hilft künstliche Intelligenz, bessere optimale Lösungen für Erdbeobachtungsaufgaben mit großen Datensätzen zu finden. Entscheidungsträger und politische Entscheider nutzen diese Lösungen bereits ausführlich, um schnelle sicherheitskritische sowie menschenzentrierte Entscheidungen zu treffen.

Dabei verspricht maschinelles Lernen mit Quantencomputern, basierend auf Quantenalgorithmen, einige Aufgaben schneller zu lösen als es datengesteuerte konventionelle Alternativen erlauben. Wir versuchen hier, verschiedene Referenzdatensätze von Satellitendaten zu nutzen, um Vorgehensweisen für maschinelles Lernen auf Quantencomputern mithilfe von traditionellen Modellen der künstlichen Intelligenz zu entwickeln und zu bewerten. Hier gibt es drei Hauptaufgaben bei der Verarbeitung von maschinellem Lernen auf Quantencomputern mit Referenzdatensätzen von Satelliten: 1. Welche Aufgaben des maschinellen Lernens für große Datensätze von Satelliten und welche datengesteuerten Aufgaben von Satelliten kann man effizient und tatsächlich auf einem Quantencomputer berechnen? 2. Wie kann man hochdimensionale Datenpunkte von Satelliten in Eingabe-Quantenzustände einbetten? sowie 3. Wie kann man sowohl Supercomputer als auch Quantencomputer gemeinsam profitabel ausnutzen?

Um eine wissenschaftlich fundierte Antwort auf diese drei Fragen zu erhalten, untersuchen und identifizieren wir sowohl Aufgaben des maschinellen Lernens als auch datengesteuerte Aufgaben von Satelliten, die auf einen Quantencomputer verteilt werden können und sonst von sich aus unlösbar wären. Danach schlagen wir unsere Encoding-Strategie für klassische Aufgabenstellungen mit großen Datensätzen von Satelliten für einen Quantencomputer vor, nämlich ein zweistufiges Encoding. Darüber hinaus entwerfen und untersuchen wir Quantenlernansätze für einen Quantenannealer und einen störanfälligen mittelgroßen Quantencomputer für Aufgaben mit überwachtem Lernen.

Für Aufgaben mit überwachtem Lernen ist die Leistung unserer Quantenansätze für maschinelles Lernen bereits konkurrenzfähig (und für einige Fälle sogar besser) als klassische Vergleichsansätze. Zusätzlich schätzen wir die benötigten Quantenressourcen, die nötig sind, um besser zu sein als ein Supercomputer und um von einem kombinierten Supercomputer mit Quantencomputer zu profitieren. Dies gibt uns Einblicke in zukünftige fehlertolerante Quantencomputer, um praktische Berechnungsaufgaben anzugehen.

Contents

Abstract	iv
Kurzfassung	v
1 Introduction	1
1.1 Earth Observation Datasets	1
1.2 The Power of Quantum Computers	2
1.3 Research Questions: Quantum Computing for Earth Observation	3
1.4 Contributions	4
1.5 Outline	5
1.6 List of publications included in this dissertation	5
2 Background	9
2.1 Artificial Intelligence	9
2.1.1 Deep Learning	10
2.2 Quantum Computing	13
2.2.1 Universal Quantum Computation	17
2.3 Quantum Artificial Intelligence	18
2.3.1 Variational Quantum Algorithms	18
2.4 Quantum Annealing	20
2.4.1 Quantum Annealing Device	21
2.5 Earth Observation	21
2.5.1 Benchmark Satellite Datasets	22
2.6 Brief Summary for Applications of Faulty Quantum Computers	24
3 Methodology	26
3.1 Hybrid Classical-Quantum Approach	26
3.1.1 Workflow Details for Hybrid Classical-Quantum Approaches	27
3.1.2 Prospects and Bottlenecks	28
3.2 Quantum Annealing Approach	28
3.2.1 Workflow Details for a Quantum Annealing Approach	29
3.2.2 Prospects and Bottlenecks	30
4 Main Results I: Gaining Insights into Small-Scale Faulty Quantum Computers	31
4.1 A NISQ Computer for Earth Observation	31
4.1.1 Publication A: Classification of Remote Sensing Images with Parameterized Quantum Gates	32
4.1.2 Publication B: Quantum Transfer Learning for Real-World, Small, and High-Dimensional Remotely Sensed Datasets	38
4.1.3 Publication C: Natural Embedding of the Stokes Parameters of Polarimetric Synthetic Aperture Radar Images in a Gate-based Quantum Computer	47

4.2	A D-Wave Quantum Annealer for Earth Observation	56
4.2.1	Publication D: A Quantum Annealer for Network Flow Minimization in InSAR Images	57
4.2.2	Publication E: A Quantum Annealer for Subset Feature Selection and the Classification of Hyperspectral Images	62
4.2.3	Publication F: Assembly of a Coreset of Earth Observation Images on a Small Quantum Computer	72
5	Main Results II: Toward Quantum Advantage	86
5.1	The Quantum Computing Assessment	86
5.1.1	Publication G: Quantum-inspired Tensor Network for Earth Science .	87
5.1.2	Publication H: Exploiting the Quantum Advantage for Satellite Image Processing: Review and Assessment	92
5.1.3	Publication I: Quantum Computing for Climate Change Detection, Climate Modeling, and Climate Digital Twins	102
6	Conclusions and Future Works	138
	Bibliography	140
	Acronyms	150
	List of Figures	152
	List of Tables	153

1

Introduction

Satellite datasets provide valuable information about changes in the Earth’s surface to end users, a field known as **Earth Observation (EO)**. They help answer scientific questions such as land cover changes, climate change impact, and hazard assessment [W1]. **Artificial Intelligence (AI)** is extensively employed to extract informative information from unseen satellite images thanks to ubiquitous benchmark datasets and supercomputing resources [23]. The heterogeneity of satellite datasets makes it challenging to analyze and recognize specific patterns even using conventional **AI** models, some of them are intractable, e.g., **Support Vector Machines (SVM)** [51].

Quantum computers are new computing devices with significantly greater computational power than traditional supercomputers [6, 62]. They use qubits and quantum gates as their fundamental components, while supercomputers use bits and logic gates at a physical level. To perform a function using a quantum computer, quantum gates are arranged in a specific order according to a set of instructions, and then they operate on input qubits.

Currently, available quantum computers are designed to address specific types of computational problems like a quadratic optimization problem [28, 107]. The search for potential applications for quantum computers is an active area of research. **Quantum Machine Learning (QML)** has emerged as a particularly promising area, indicating computational advantages over some traditional **AI** models [11]. It represents the fusion of **Quantum Computing (QC)** and **AI**, aiming to efficiently and effectively process, e.g., satellite datasets than conventional **AI** models deployed on supercomputers [66].

This chapter aims to provide a high-level overview of why quantum computing is suitable for some **EO** problems, presents our main findings, and outlines the structure of this thesis. Toward quantum advantage in practical **EO** challenges, the main goal of this dissertation can be summarized in the quote: “we have to learn to walk before we can run” by E.L. James.

1.1 Earth Observation Datasets

Satellite and airborne platforms use sensors to measure back-scattered signals of different wavelengths on the Earth’s surface. These signals carry information about the physical properties of target objects, such as their phase, intensity, spectral band, and polarization [84]. Scientists have been using real-world satellite datasets in order to mitigate complex environmental and climate challenges [77]. To tackle these challenges, research studies combine practical **EO** methodologies with satellite datasets such as **AI** models for processing **Synthetic Aperture Radar (SAR)** datasets [108], integer programming for **SAR** tomography [91], and Bayesian analysis for quantifying uncertainties in satellite datasets [19, 98].

Nowadays, many **EO** challenges involve big datasets from various sources, such as dedicated experiments, and distinct sensor devices. Thanks to advances in modern supercomputing resources and benchmark satellite datasets, it is possible to detect ground targets and analyze changes on the Earth’s surface using large **AI** models like **Machine Learning (ML)** and **Deep Learning (DL)** [13, 61, 100, 108].

Some of the benchmark satellite datasets include images from the [Interferometric Synthetic Aperture Radar \(InSAR\)](#) technique [26], Eurosat [43], UC Merced Land Use dataset [23], [Hyperspectral Image \(HSI\)](#) datasets of Indian Pine and Pavia University [W2], and [Polarimetric Synthetic Aperture Radar \(PolSAR\)](#) images of California and Oberpfaffenhofen [52]. Chapter 2.5 presents these benchmark satellite datasets in a more detail.

We note that processing satellite datasets using [AI](#) models is much more complex and harder than typical red-green-blue image processing due to their heterogeneity and modality [51]. Decision-makers and policymakers use solutions of these methods to make safety-critical and human-centered choices [23].

1.2 The Power of Quantum Computers

The world's most powerful supercomputers are capable of solving a wide range of hard computational problems in academia and industry. However, there is a specific class of computational problems for which no efficient classical algorithms are known [6, 62]. Some computational problems are notoriously difficult, even for supercomputers. This has motivated a number of scientists to study and employ quantum algorithms [9, 32, 57, 79].

To run practical quantum algorithms, hundreds of qubits are required. For instance, executing a specific quantum algorithm may require $n = 100$ qubits, resulting in an n -qubit state composed of 2^{100} complex amplitudes that need to be tracked. To store 2^{100} complex amplitudes, we need at least 10^{16} petabytes of memory if each complex number is represented by 2×128 bits. The supercomputer located at the Leibniz Supercomputing Centre (LRZ) in Munich, Germany, has around 0.719 petabytes of main memory, and even the best current supercomputer in the world, located at the Oak Ridge National Laboratory in Tennessee, USA, has only around 10 petabytes of main memory [97]. Hence, it is impossible to deploy quantum algorithms on the currently available supercomputers since quantum algorithms for real-world problems of practical relevance would require hundreds of qubits [80].

To resolve this bottleneck, some organizations start to offer a quantum computer, e.g, a D-Wave systems and International Business Machines (IBM) Corporation. These quantum computers are divided generally into two categories: digital quantum computers and analog quantum simulators as envisioned by some physicists such as Richard Feynman, Yuri Manin, and Paul Benioff [9, 32, 57].

Quantum computers also promise to accelerate some computational algorithms according to computational complexity theorems/conjectures [2, 11, 27, 89]. Computational complexity theorems draw boundaries between computational problems according to their hardness or the computational time required. They are governed by currently well-known rules: [Polynomial \(P\)](#) \neq [Non-Deterministic Polynomial \(NP\)](#), $P \subset NP$, and $P \subset$ [Bounded-Error Quantum Polynomial \(BQP\)](#). The hardness of computational problems is illustrated in Figure 1.1. Here, [P](#) class comprises efficiently solvable computational problems on a classical computer in a reasonable polynomial time. [NP](#) class includes hard computational problems whose solutions cannot be found efficiently on a classical computer in a reasonable polynomial time but can be verified by solutions given by some oracle on a classical computer in a reasonable polynomial time. [Non-Deterministic Polynomial Complete \(NP-complete\)](#) class is the hardest computational problem in the [NP](#) class, and the [BQP](#) class includes hard computational problems on a classical computer but can be solved efficiently by polynomial quantum circuits with a $2/3$ probability of success on a quantum computer [6].

Quantum circuit models in the [BQP](#) class can efficiently solve some notoriously hard computational problems contained in the [NP](#) class on a quantum computer [1, 101], such as Shor's algorithm for factoring a large number [93]. Moreover, some quantum learning models

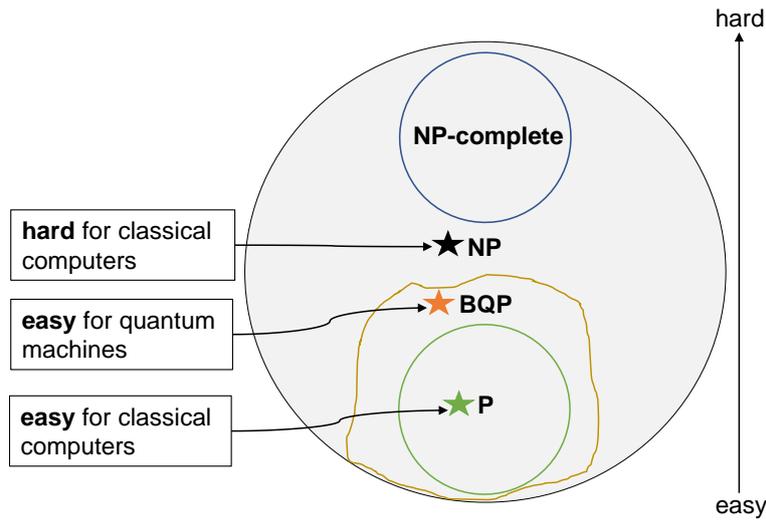


Figure 1.1: A computational complexity conjecture draws boundaries between computational problems according to their hardness based on the required classical and quantum computational resources. In particular, the computational problem denoted by the **green star** is easy for both a quantum computer and a classical computer, the computational problem denoted by the **orange star** is easy for a quantum computer but hard for a classical computer. The computational problem denoted by the **black star** is hard for a classical computer, but we are not aware of any existing quantum algorithmic approaches.

like QML deployed on a quantum computer are (exponentially) faster and more efficient than their conventional counterpart models [6, 29, 56, 62, 93]. For example, a Quantum Support Vector Machine (QSVM) shows a polynomial speed-up over its classical counterpart [81], and a Quantum Neural Network (QNN) outperforms its conventional counterpart Convolutional Neural Network (CNN). Refer to the website [W3] on conventional methods boosted by quantum algorithms.

1.3 Research Questions: Quantum Computing for Earth Observation

Current quantum computers available on the market, such as Quantum Annealing (QA) devices and Noisy Intermediate-Scale Quantum (NISQ) computers, have limited error-prone input qubits and quantum gates. They are designed to solve specific kinds of computational problems, for example, a QA device, for tackling Quadratic Unconstrained Binary Optimization (QUBO) problems. On contrary, EO challenges involving large-scale satellite datasets require high spatial dimensionality and floating-point numbers, thereby limiting the embedding of EO methodologies like an SVM for land cover changes in the input qubits [66].

It is essential to address how much computational advantage can we gain from QML models with the help of traditional classical resources over conventional AI models when considering current and future quantum computers? To answer this question we tackle three research conundrums realizing the desired potential quantum advantage of a quantum computer for processing big satellite datasets:

1. which machine learning task for satellite datasets and which satellite data-driven task must we employ to understand currently existing quantum computers and to gain

insights into possible future quantum computers?

2. how can satellite data points be embedded in quantum computers?
3. how can we profit from both supercomputers and (future) quantum computers?

This thesis identifies satellite datasets and corresponding machine learning models posed as **QML** approaches. We investigate the potential advantages and imperfections of **QML** models for object recognition tasks in **EO**. Additionally, we look into innovative methods to encode satellite datasets into a small number of error-prone qubits. We design and benchmark task-driven **QML** approaches with traditional classical methods. We then present how to process benchmark satellite datasets on a **QA** device and a **NISQ** computer. We also aim to gain insight into a future **Fault-Tolerant Quantum Computer (FTQC)** for data-driven computational problems. The key findings of this thesis are centered around the use of **QML** models deployed on small-scale faulty quantum computers for processing large-scale satellite datasets for detecting changes on the Earth’s surface. Hence, this dissertation is titled “*Quantum Machine Learning for Large-Scale Classical Datasets with Applications in Earth Observation*” and is based on a series of publications by this author.

1.4 Contributions

In our approach, we utilized both a D-Wave quantum annealer and a small-scale, faulty digital quantum computer. The D-Wave quantum annealer functions as an analog quantum simulator, while the small-scale faulty digital quantum computer, known as a **NISQ** computer, features a limited number of imperfect qubits and quantum gates, making them faulty quantum computers. Our research explored various machine learning models implemented on these faulty quantum computers to process satellite datasets.

- For the **QA** device offered by D-Wave systems, we design **QML** approaches for satellite datasets such as a **QSVM** and a feature selection algorithm. They belong to an **NP** class. We benchmark the performances of our proposed methods compared to those generated by classical **ML** techniques. For these tasks, we discovered that the D-Wave quantum annealer performs similarly to conventional methods and classical annealers, and even outperforms them in some cases [64, 65].
- For the **NISQ** computer, we design a hybrid classical-quantum model using a **Parameterized Quantum Circuit (PQC)** model with the aid of conventional **AI** techniques and a classical computer. We also propose a so-called two-level encoding for embedding data points in qubits. Our two-level encoding technique and training of **PQCs** on high-dimensional satellite datasets are called a hybrid classical-quantum model that even promises computational advantage over some conventional **AI** models. Our hybrid approach is more specifically compatible with **NISQ** computers for satellite datasets. It tackles two main challenges encountered with **NISQ**-era computers [67, 75]: 1. encoding satellite datasets in **NISQ** computers, and 2. benchmarking **QML** methods against each other and conventional **AI** methods. We have found that the performance of our hybrid approach on big satellite datasets is competitive with their classical counterparts [66].

The scientific answers presented in this thesis provide significant advances for applying small quantum computers to **EO** data-driven tasks. The findings presented here will aid in designing **QML** models with potential quantum advantages for solving **EO** data-driven

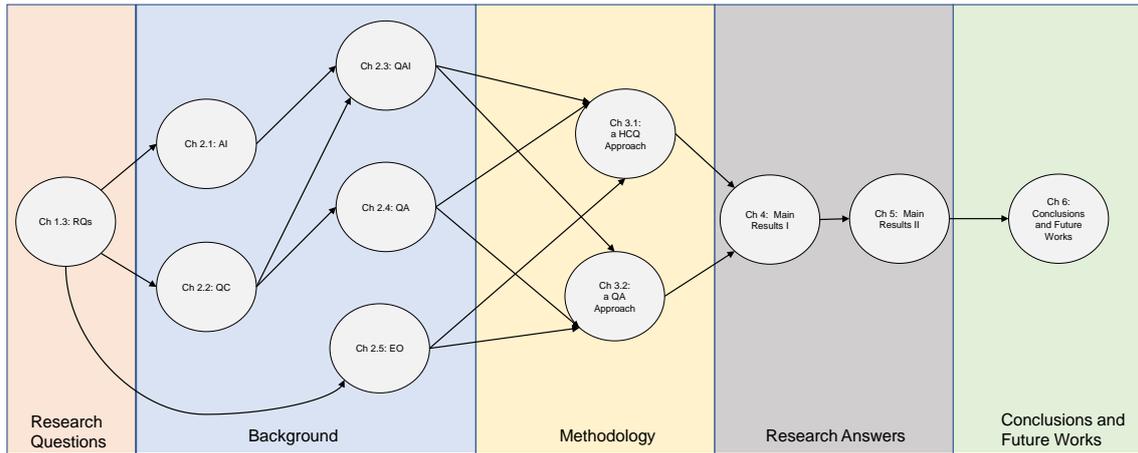


Figure 1.2: The general causal structure of this thesis comprises: Chapter 1.3: Research Questions (RQs), Chapter 2: Background including Artificial Intelligence (AI), Quantum Computing (QC), Quantum Artificial Intelligence (QAI), Quantum Annealing (QA), and Earth Observation (EO) datasets, Chapter 3: Methodology comprising a Hybrid Classical-Quantum Approach (HCQ Approach) and a Quantum Annealing Approach (QA Approach), Chapter 4: Main Results I, Chapter 5: Main Results II, and Chapter 6: Conclusions and Future Work.

problems of practical importance, and broaden our perspectives on integrating a future quantum computer with a conventional classical computer. More importantly, this thesis extends existing computational methodologies for challenging EO problems and beyond while also providing new insights into the potential of a future quantum computer.

1.5 Outline

The following text outlines the structure of this thesis. In Chapter 2, we cover the basic building blocks of AI, QA and QA, introduce real-world EO datasets, and applications of hybrid approaches. Chapter 3 discusses the methodology for answering our research questions, and the prospects and bottlenecks of small-scale faulty quantum computers. In Chapter 4, titled “Main Results I: gaining insights into small-scale faulty quantum computers,” and Chapter 5, titled “Main Results II: toward quantum advantage,” we provide research answers and main results based on several peer-reviewed publications. Finally, Chapter 6 concludes this dissertation. We present the general causal structure of this dissertation in Figure 1.2.

1.6 List of publications included in this dissertation

- **Publication A:** S. Otgonbaatar and M. Datcu, “Classification of Remote Sensing Images with Parameterized Quantum Gates,” in *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1-5, 2022, art no. 8020105, URL: <https://doi.org/10.1109/LGRS.2021.3108014> [66].
 - *CRedit*.¹ S. Otgonbaatar: conceptualization, methodology, software, validation, formal analysis, investigation, writing - original draft, writing - reviewing and editing, visualization, M. Datcu: supervision.

¹The authors contribution is based on a Contributor Roles Taxonomy (CRedit) system: <https://credit.niso.org/>

- *Own contributions.* S. Otgonbaatar conceptualized, performed numerical experiments, wrote the article, and finalized for the publication throughout the review process.
- *Other contributors.* M. Datcu supervised.
- **Publication B:** S. Otgonbaatar, G. Schwarz, M. Datcu, and D. Kranzlmüller, “Quantum Transfer Learning for Real-World, Small, and High-Dimensional Remotely Sensed Datasets,” in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 16, pp. 9223-9230, 2023, doi: 10.1109/JSTARS.2023.3316306, URL: <https://doi.org/10.1109/JSTARS.2023.3316306> [75].
 - *CRedit.* S. Otgonbaatar: conceptualization, methodology, software, validation, formal analysis, investigation, writing - original draft, writing - reviewing and editing, visualization, G. Schwarz: writing - reviewing and editing, M. Datcu: supervision, D. Kranzlmüller: supervision.
 - *Own contributions.* S. Otgonbaatar conceptualized, performed numerical experiments, wrote the article, and finalized for the publication throughout the review process.
 - *Other contributors.* G. Schwarz helped improve the quality of the text. M. Datcu and D. Kranzlmüller supervised.
- **Publication C:** S. Otgonbaatar and M. Datcu, “Natural Embedding of the Stokes Parameters of Polarimetric Synthetic Aperture Radar Images in a Gate-based Quantum Computer,” in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1-8, 2022, art no. 4704008, doi: 10.1109/TGRS.2021.3110056, URL: <https://doi.org/10.1109/TGRS.2021.3110056> [67].
 - *CRedit.* S. Otgonbaatar: conceptualization, methodology, software, validation, formal analysis, investigation, writing - original draft, writing - reviewing and editing, visualization, M. Datcu: conceptualization, supervision.
 - *Own contributions.* S. Otgonbaatar conceptualized, performed numerical experiments, wrote the article, and finalized for the publication throughout the review process.
 - *Other contributors.* M. Datcu supervised.
- **Publication D:** S. Otgonbaatar and M. Datcu, “Quantum Annealer for Network Flow Minimization in InSAR Images,” *EUSAR 2021; 13th European Conference on Synthetic Aperture Radar*, Leipzig, Germany, 2021, pp. 1-4, URL: <https://ieeexplore.ieee.org/document/9472581> (Virtual Symposium) [68].
 - *CRedit.* S. Otgonbaatar: conceptualization, methodology, software, validation, formal analysis, investigation, writing - original draft, writing - reviewing and editing, visualization, M. Datcu: conceptualization, supervision.
 - *Own contributions.* S. Otgonbaatar conceptualized, performed numerical experiments, wrote the article, and finalized for the publication throughout the review process.
 - *Other contributors.* M. Datcu supervised.
- **Publication E:** S. Otgonbaatar and M. Datcu, “A Quantum Annealer for Subset Feature Selection and the Classification of Hyperspectral Images,” in *IEEE Journal of*

Selected Topics in Applied Earth Observations and Remote Sensing, vol. 14, pp. 7057-7065, 2021, doi: 10.1109/JSTARS.2021.3095377, URL: <https://doi.org/10.1109/JSTARS.2021.3095377> [64].

- *CRedit*. S. Otgonbaatar: conceptualization, methodology, software, validation, formal analysis, investigation, writing - original draft, writing - reviewing and editing, visualization, M. Datcu: supervision.
 - *Own contributions*. S. Otgonbaatar conceptualized, performed numerical experiments, wrote the article, and finalized for the publication throughout the review process.
 - *Other contributors*. M. Datcu supervised.
- **Publication F:** S. Otgonbaatar and M. Datcu, “Assembly of a Coreset of Earth Observation Images on a Small Quantum Computer,” *Electronics* 10, no. 20: 2482., 2021, URL: <https://doi.org/10.3390/electronics10202482> [65].
 - *CRedit*. S. Otgonbaatar: conceptualization, methodology, software, validation, formal analysis, investigation, writing - original draft, writing - reviewing and editing, visualization, M. Datcu: supervision.
 - *Own contributions*. S. Otgonbaatar conceptualized, performed numerical experiments, wrote the article, and finalized for the publication throughout the review process.
 - *Other contributors*. M. Datcu supervised.
- **Publication G:** S. Otgonbaatar and D. Kranzlmüller, “Quantum-inspired Tensor Network for Earth Science,” *IEEE International Geoscience and Remote Sensing Symposium*, Pasadena, CA, USA, 2023, pp. 788-791, URL: <https://doi.org/10.1109/IGARSS52108.2023.10282577> [73].
 - *CRedit*. S. Otgonbaatar: conceptualization, methodology, software, validation, formal analysis, investigation, writing - original draft, writing - reviewing and editing, visualization, D. Kranzlmüller: supervision.
 - *Own contributions*. S. Otgonbaatar conceptualized, performed numerical experiments, wrote the article, and finalized for the publication throughout the review process.
 - *Other contributors*. D. Kranzlmüller supervised.
- **Publication H:** S. Otgonbaatar and D. Kranzlmüller, “Exploiting the Quantum Advantage for Satellite Image Processing: Review and Assessment,” in *IEEE Transactions on Quantum Engineering*, vol. 5, pp. 1-9, 2024, art no. 3100309, doi: 10.1109/TQE.2023.3338970, URL: <https://doi.org/10.1109/TQE.2023.3338970> [72].
 - *CRedit*. S. Otgonbaatar: conceptualization, methodology, software, validation, formal analysis, investigation, writing - original draft, writing - reviewing and editing, visualization, D. Kranzlmüller: supervision.
 - *Own contributions*. S. Otgonbaatar conceptualized, performed numerical experiments, wrote the article, and finalized for the publication throughout the review process.
 - *Other contributors*. D. Kranzlmüller supervised.

- **Publication I:** S. Otgonbaatar, O. Nurmi, M. Johansson, and et. al., “Quantum Computing for Climate Change Detection, Climate Modeling, and Climate Digital Twins,” 2023, *project report*, URL: <http://dx.doi.org/10.36227/techrxiv.24478663.v1> [74].
 - *CRedit*. S. Otgonbaatar: conceptualization, methodology, writing - original draft, writing - reviewing and editing, O. Nurmi: writing - original draft, M. Johansson: writing - original draft, J. Mäkelä: writing - original draft, T. Kocman: writing - original draft, P. Gawron: writing - original draft, Z. Puchala: writing - original draft, J. Mielczarek: writing - original draft, A. Miroszewski: writing - original draft, C.O. Dumitru: supervision.
 - *Own contributions*. S. Otgonbaatar conceptualized, performed numerical experiments, wrote the “quantum uncertainty quantification” section and the main text in the article, and finalized for the publication.
 - *Other contributors*. Other authors wrote their respective sections. O. Nurmi wrote the main text on quantum computing, and M. Johansson wrote “quantum for climate modeling” section. All other authors contributed equally for writing this article.

List of publications not included in this dissertation

- S. Otgonbaatar, “Quantum Digital Twins,” 2024, in preparation for a *journal*, URL: <https://github.com/sozoluffy/Quantum-Digital-Twins/tree/main> [W4].
- S. Otgonbaatar, M. Datcu and B. Demir, “Coreset of Hyperspectral Images on a Small Quantum Computer,” *IEEE International Geoscience and Remote Sensing Symposium* (Virtual Presentation), Kuala Lumpur, Malaysia, 2022, pp. 4923-4926, doi: 10.1109/IGARSS46834.2022.9884273, URL: <https://doi.org/10.1109/IGARSS46834.2022.9884273> [71].
- S. Otgonbaatar, M. Datcu, and B. Demir, “Causality for Remote Sensing: An Exploratory Study,” *IEEE International Geoscience and Remote Sensing Symposium* (Virtual Presentation), Kuala Lumpur, Malaysia, 2022, pp. 259-262, doi: 10.1109/IGARSS46834.2022.9883060, URL: <https://doi.org/10.1109/IGARSS46834.2022.9883060> [70].
- S. Otgonbaatar and M. Datcu, “Quantum Annealing Approach: Feature Extraction and Segmentation of Synthetic Aperture Radar Image,” *IEEE International Geoscience and Remote Sensing Symposium* (Virtual Presentation), 2020, pp. 3692-3695, doi: 10.1109/IGARSS39084.2020.9323504, URL: <https://ieeexplore.ieee.org/document/9323504> [69].

2

Background

AI involves the study of learning algorithms for pattern recognition tasks deployed on a classical computer, whereas **Quantum Artificial Intelligence (QAI)** is the synergy between AI and QC for extracting informative patterns from benchmark datasets (in our case, benchmark EO datasets) deployed on a quantum computer [11]; here, QAI is exchangeable with QML.

Variational Quantum Algorithms (VQAs) and QA approaches are a kind of QAI specifically designed for NISQ computers and QA devices, respectively, with the help of classical computing methods and a conventional classical computer [21]. Hence, they are sometimes called hybrid classical-quantum approaches, the synergy between quantum computing and classical computing.

This chapter provides background on AI, QC, QAI, QA, and EO datasets. In Chapter 2.6, we provide brief summaries of processing hybrid classical-quantum approaches on satellite datasets when deploying a quantum part on faulty quantum computers, NISQ and QA devices. Chapter 3 presents the methodologies of hybrid classical-quantum and QA approaches, and how to deploy them on both a NISQ computer and a D-Wave QA device for processing big satellite datasets.

Several books [13, 34, 61] provide a more detailed discussion of the various methodologies of AI models, and the book [84] provides a discussion of satellite datasets for EO. We refer to [8, 62] for an interested reader for the detailed concepts of quantum computing and quantum information science.

2.1 Artificial Intelligence

AI generally refers to learning methods to extract informative features and patterns from benchmark datasets [13, 61]. In particular, the set of learning methods includes Bayesian Inference (BI) [24, 88], SVM [60, 87], Boltzmann Machines (BM) [86], Deep Neural Networks (DNN), or Artificial Neural Networks (ANN) [3, 34]. One can classify, cluster, and predict unseen similar datasets by leveraging benchmark datasets. For simplicity, a set of learning methods can be represented by a parameterized model $f_{\mathbf{w}}$, where \mathbf{w} is its adjustable weights. We can divide it into two distinct learning paradigms: supervised and unsupervised learning [13, 61].

Supervised learning is a method for recognizing hidden patterns from benchmark datasets, which are denoted by $\mathcal{S} = \{\mathbf{x}_j, y_j\}_{j=1}^N$. In this context, \mathbf{x}_j represents a training data point, and y_j represents its corresponding label. It is assumed that a benchmark dataset is governed by some model $y_j = f_{\mathbf{w}^*}(\mathbf{x}_j)$, where \mathbf{w}^* is the hidden-known generating dataset's label y_j . Moreover, supervised learning methods intend to learn the model $\tilde{y}_j = f_{\mathbf{w}}(\mathbf{x}_j)$, where \mathbf{w} is the adjustable-unknown, such that the loss function between a given label y_j and a predicted label \tilde{y}_j is minimum at some value $\tilde{\mathbf{w}}$:

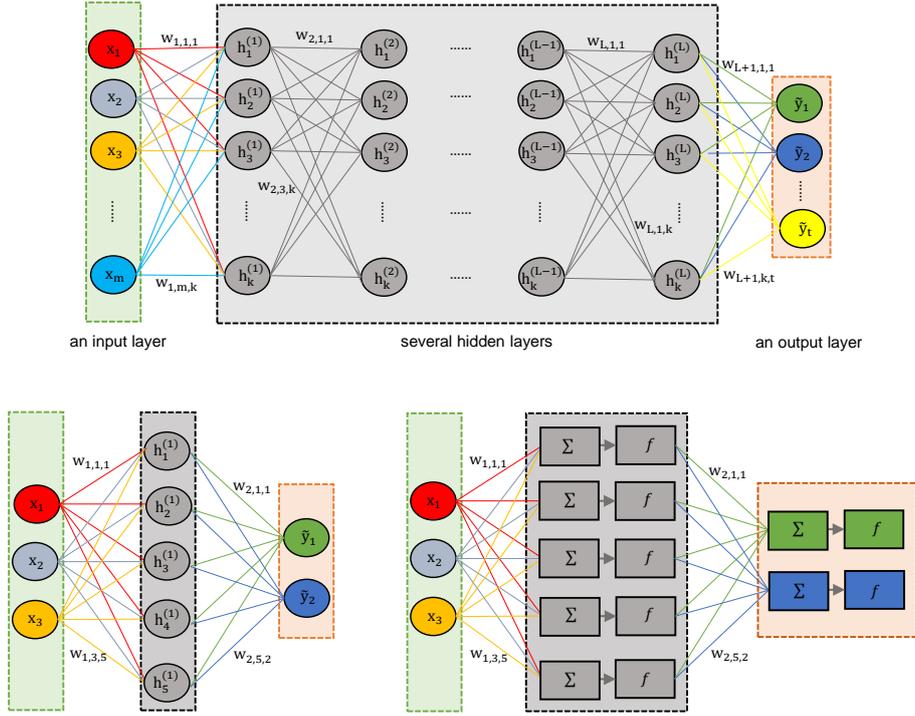


Figure 2.1: [Top] An ANN comprises an input layer, several hidden layers, and an output layer. It is worth noting that the number of neurons in each hidden layer k does not have to be the same. In most cases, the number of input points m is also greater than the number of output points t . [Bottom Left] We present an example ANN with a single hidden layer. [Bottom Right] Each digital value of both the hidden and output layer is obtained by summing the incoming information Σ and applying a nonlinear activation function f on the summed result: the neuron in the hidden layer (gray), and the output neuron (green and blue).

$$\begin{aligned} \tilde{\mathbf{w}} &= \arg \min_{\mathbf{w}} \mathcal{L}(f_{\mathbf{w}}(\mathbf{x}_j), f_{\mathbf{w}^*}(\mathbf{x}_j)) \\ &= \arg \min_{\mathbf{w}} \mathcal{L}_{\mathbf{w}}(\tilde{y}_j, y_j), \end{aligned} \tag{2.1}$$

where $\mathcal{L}_{\mathbf{w}}(f_{\mathbf{w}}(\mathbf{x}_j), f_{\mathbf{w}^*}(\mathbf{x}_j))$ is the so-called loss function. We choose the loss function depending on the task at hand. For instance, the loss function can be a squared loss if the task is a dataset regression, and an entropy loss if the task is a dataset classification.

Unsupervised learning is another learning paradigm for recognizing hidden patterns from a benchmark dataset denoted by $\mathcal{S} = \{\mathbf{x}_j\}_{j=1}^N$ when the label information y_j is not provided. In particular, it is a machinery of discovering the underlying distribution function $p(\mathbf{w}|\mathcal{S})$ of data points:

$$\tilde{\mathbf{w}} = \arg \max_{\mathbf{w}} p(\mathbf{w}|\mathcal{S}) \tag{2.2}$$

Two examples of unsupervised learning are dimensionality reduction and clustering of data points $\mathcal{S} = \{\mathbf{x}_j\}_{j=1}^N$.

2.1.1 Deep Learning

DL refers to deep neural networks for extracting underlying informative patterns from benchmark datasets (in our case, EO images), similar to traditional artificial neural networks [53].

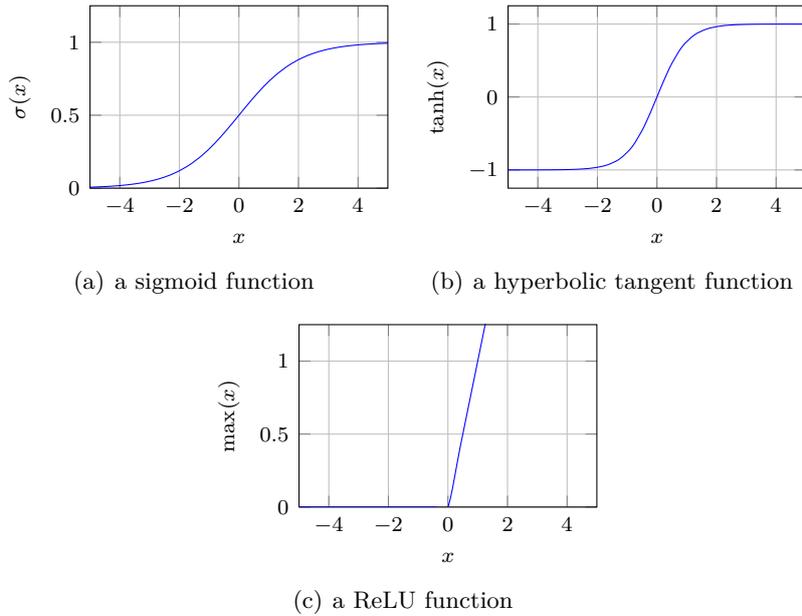


Figure 2.2: Non-linear activation functions include the logistic sigmoid, $\sigma(x)$, the hyperbolic tangent, $\tanh(x)$, and the **ReLU** function, $\max(0, x)$.

In particular, it extracts macroscopic patterns from images using filters with a few trainable weights in contrast to artificial neural networks [50]. Deep neural networks are mostly called a **Convolutional Network (ConvNet or CNN)**. Several different **CNN** architectures, including ResNet [42] and VGG16 [94], exist for analyzing benchmark datasets. More importantly, we can use **CNNs** for both supervised and unsupervised learning tasks.

In the following subsections, we present a high-level overview of **ANNs** and **CNNs** for supervised learning. For a deeper insight into supervised and unsupervised learning paradigms, the reader is referred to [13, W5, 61].

An Artificial Neural Network

An **ANN** is a digital information processing technique inspired by the goal of modeling a biological neural system [17, 44]. It comprises layers of artificial neurons that process information similarly to biological neurons in the human brain. **ANNs** consist of three groups of layers: an input layer, an output layer, and several hidden layers. Each neuron in a layer is connected to all the neurons in its nearest neighbor layer via edges with trainable weights, as shown in Figure 2.1.

Mathematically, the input layer receives benchmark dataset's features $\mathbf{x} = (x_1, x_2, \dots, x_m)^T$. And the digital values of the hidden layer $\mathbf{h}^{(l)} = (h_1^{(l)}, h_2^{(l)}, \dots, h_z^{(l)})^T$, $1 \leq l \leq L$, and the output layer $\tilde{\mathbf{y}} = (\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_t)^T$ are set using the weighted sum of digital information of their previous layer and using the nonlinear activation function f sequentially. Nonlinear activation functions mimicking excitatory and inhibitory transitions act as a threshold value for activating an artificial neuron given its summation value.

Figure 2.2 illustrates commonly used nonlinear activation functions: the sigmoid function, $\sigma(x) = 1/(1+e^x)$, the hyperbolic tangent, $\tanh(x) = (e^x + e^{-x})/(e^x - e^{-x})$, and the **Rectified Linear Unit (ReLU)** function, $\text{ReLU}(x) = \max(0, x)$.

An example **ANN** shown in Figure 2.1 [Bottom] consists of the input layer with three digital values $\mathbf{x}_j = (x_1, x_2, x_3)^T$, the single hidden layer with five digital values $\mathbf{h}^{(1)} =$

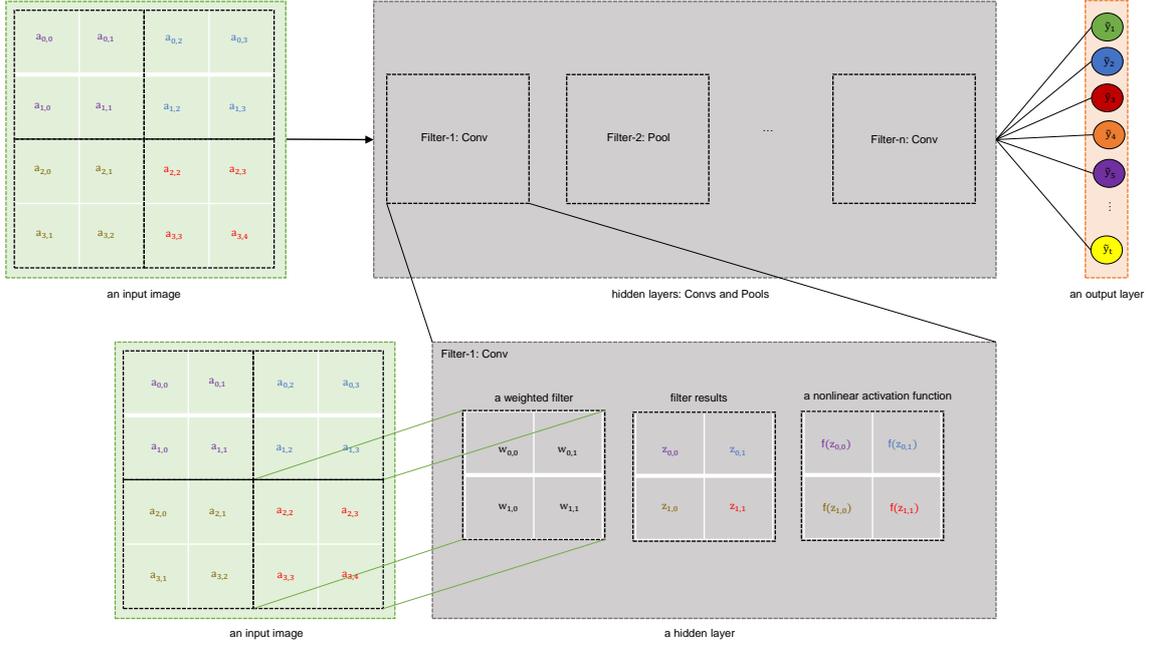


Figure 2.3: A simple CNN architecture: [Top] the filters, including Conv and Pool layers, convolve the input image and feed their result of the nonlinear activation function to their next layer, either Conv, Pool or an FCNL. [Bottom] The operation of Filter-1.

$(h_1^{(1)}, h_2^{(1)}, h_3^{(1)}, h_4^{(1)}, h_5^{(1)})^T$, and the output layer with two digital values $\tilde{\mathbf{y}} = (\tilde{y}_1, \tilde{y}_2)^T$. The digital values of the hidden and output layer are defined, respectively:

$$h_k^{(1)} = f \left(\underbrace{b_{1,k} + \sum_{j=1}^m w_{1,j,k} x_j}_{\Sigma} \right) = \max \left(0, \Sigma \right), \quad k = 1, \dots, 5 \quad (2.3)$$

$$\tilde{y}_t = f \left(\underbrace{b_{2,t} + \sum_{j=1}^k w_{2,j,t} h_j^{(1)}}_{\Sigma} \right) = \max \left(0, \Sigma \right), \quad t = 1, 2 \quad (2.4)$$

where $b_{1,k}$ and $b_{2,t}$ are biases, \mathbf{x}_j is a data point included in $\mathcal{S} = \{\mathbf{x}_j, y_j\}_{j=1}^N$, $w_{1,j,k}$ and $w_{2,j,t}$ are trainable weights, and f is the ReLU function. In most cases, the summation operation and nonlinear activation functions are not explicitly shown, but the digital values $(h_k^{(l)}, \tilde{y}_t)$.

A Convolutional Neural Network

A CNN is a type of neural network that consists of different layers of convolution operations along with nonlinear activation functions [42, 94]. CNN architectures like ResNet [42], VGG16 [94], AlexNet [48], Inception-4 [99], and variational autoencoders [47] mainly comprise some Convolutional (Conv), Pooling (Pool), and a Fully-Connected Neuron Layer (FCNL) that represents the output layer. Figure 2.3 [Top] depicts the architecture of a

typical CNN. The CNN's convolution and pooling layers are essentially weighted filters that capture different informative features like vertical edges, circles, and other relevant information from images [48, 105]. The weighted filters convolve sequentially over the raw image, and their weights capture specific characteristics of the input image, while the nonlinear activation functions threshold the filter results in particular ranges.

For instance, suppose we have a 4×4 input image with a single channel and a 2×2 weighted filter. The filter results are obtained by a dot product of the input image's 2×2 elements and the weighted filter, as shown in Figure 2.3 [Bottom].

$$\begin{aligned} z_{0,4} &= b_0 + a_{1,1}w_{0,1} + a_{1,2}w_{0,2} + a_{1,5}w_{0,3} + a_{1,6}w_{0,4} \\ &= \mathbf{x}^T \mathbf{w} = \mathbf{w}^T \mathbf{x}, \end{aligned} \quad (2.5)$$

where b_0 is the bias term, $\mathbf{x} = (1, a_{1,1}, a_{1,2}, a_{1,5}, a_{1,6})^T$, $\mathbf{w} = (b_0, w_{0,1}, w_{0,2}, w_{0,3}, w_{0,4})^T$, and $z_{0,1}, z_{0,2}$, and $z_{0,3}$ are computed in a similar way to $z_{0,4}$. The ReLU activation function is calculated as $f(z_{0,4}) = \max(0, z_{0,4})$. Here, *red* denotes the sample calculation of the hidden layer, the bottom right corner of an input image illustrated in Figure 2.3 [Bottom].

As explained, we assume an image dataset denoted by $\mathcal{S} = \{\mathbf{x}_j, y_j\}_{j=1}^N$, where each image $\mathbf{x}_j \in \mathbb{R}^{m \times m}$ is associated with a label $y_j \in \mathbb{R}$. To train a CNN on this dataset and obtain its optimal weight $\tilde{\mathbf{w}}$ by minimizing the loss function $\mathcal{L}_{\mathbf{w}}(\tilde{y}_j, y_j)$ of the predicted value \tilde{y}_j and the true label y_j , one applies a widely used method for minimizing this loss function via the (stochastic) gradient descent:

$$\mathbf{w}_{new} = \mathbf{w}_{old} - \lambda \frac{\partial \mathcal{L}_{\mathbf{w}}(\tilde{y}_j, y_j)}{\partial \mathbf{w}}, \quad (2.6)$$

where λ is a learning rate, and the gradient $\frac{\partial \mathcal{L}_{\mathbf{w}}(\tilde{y}_j, y_j)}{\partial \mathbf{w}}$ is obtained by using backpropagation.

CNN models have been more successful for data-driven tasks than traditional computational algorithms. However, there is still room for improvement in their computational speed and output accuracy, even in their electric power usage. Hence, some scientists have attempted to increase the computational speed of conventional machine learning tasks by leveraging QC [11]. For example, they have developed quantum least squares fitting, known as a Harrow-Hassidim-Lloyd (HHL) algorithm [39], and the QNN [2, 11] much faster than their conventional counterparts.

In the following chapters, we introduce QC and AI boosted by QC (for short, QAI). We also present satellite datasets used for deploying QAI models on faulty quantum computers.

2.2 Quantum Computing

A quantum computer is a novel computing system that operates on the principles of quantum computing. Its basic building blocks are qubits and quantum gates. Quantum computing involves four fundamental concepts [62]: *qubits* that encode input data, *qubit entanglement and superposition* that generate entangled quantum states presenting a non-classical physical property, *parameterized and non-parameterized quantum gates* that manipulate the initialized qubits, and *measurement* finally generates the results. We formally define these concepts:

1. *qubits* are represented by ket vectors, denoted as $|0\rangle$ and $|1\rangle$, where $|0\rangle$ is equal to $\begin{pmatrix} 1 \\ 0 \end{pmatrix}$ and $|1\rangle$ is equal to $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$. Both $|0\rangle$ and $|1\rangle$ belong to the complex vector space \mathbb{C}^2 .

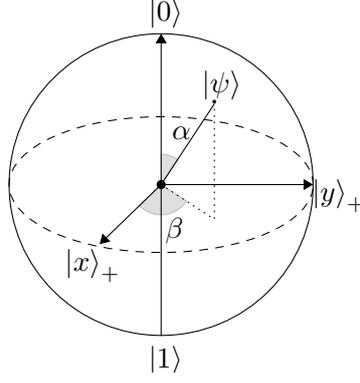


Figure 2.4: The Bloch sphere representation of a qubit.

The Hermitian of $|0\rangle$ and $|1\rangle$ are known as bra vectors, represented by $|0\rangle^\dagger = \langle 0|$ and $|1\rangle^\dagger = \langle 1|$, respectively. The Hermitian operation is represented by a dagger notation denoted by † , which signifies complex conjugation and transposition [8].

Quantum computing uses the concepts of superposition and entanglement, which are not present in classical computing. In classical computing, a bit can only exist in two states, either 0 or 1, with a probability of 1. However, a single-qubit state can be in superposition, i.e., a linear combination of $|0\rangle$ and $|1\rangle$, with complex coefficients a_0 and b_0 :

$$|\psi\rangle_0 = a_0 |0\rangle + b_0 |1\rangle \quad \text{s.t.} \quad |a_0|^2 + |b_0|^2 = 1, \quad a_0, b_0 \in \mathbb{C}. \quad (2.7)$$

The coefficients a_0 and b_0 represent the probability amplitudes of the qubits $|0\rangle$ and $|1\rangle$, respectively. The sum of the probability amplitudes is equal to one, i.e., $|a_0|^2 + |b_0|^2 = 1$. For an n -qubit state, there are 2^n amplitudes. The n -qubit state can be represented as

$$|\psi\rangle_n = \prod_{i=0}^{n-1} (a_i |0\rangle + b_i |1\rangle) = \sum_{i=0}^{2^n-1} c_i |i\rangle \quad (2.8)$$

where c_i represents the probability amplitude of the state $|i\rangle$, and $\sum_i |c_i|^2 = 1$.

As mentioned in Chapter 1, simulating this superposition state on a classical computer becomes extremely challenging, especially when $n = 100$, as we need to store 2^{100} complex numbers in classical memory.

A qubit can also be visualized as a point on/in the Bloch sphere [8], a spherical polar coordinate system, by using:

$$|\psi\rangle = \cos\left(\frac{\alpha}{2}\right) |0\rangle + e^{i\beta} \sin\left(\frac{\alpha}{2}\right) |1\rangle, \quad (2.9)$$

where α and γ are angles in radians. For example, a qubit is located at the surface of the Bloch sphere as illustrated in Figure 2.4:

- the qubit $|\psi\rangle = |x\rangle_+ = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ resides at the x-axis if $\alpha = \pi/2$ and $\beta = 0$.
- the qubit $|\psi\rangle = |y\rangle_+ = \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle)$ is at the y-axis if $\alpha = \beta = \pi/2$.
- the qubit $|0\rangle$ is at the north pole if $\alpha = 0$ and $\beta = 0$.

Table 2.1: Some single-qubit quantum gates

Name	Symbol	Matrix	Operation	Gate Diagram
Pauli-I	I	$\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$	$I 0\rangle = 0\rangle, I 1\rangle = 1\rangle$	
Pauli-X	X	$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$	$X 0\rangle = 1\rangle, X 1\rangle = 0\rangle$	
Pauli-Y	Y	$\begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}$	$Y 0\rangle = i 1\rangle, Y 1\rangle = -i 0\rangle$	
Pauli-Z	Z	$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$	$Z 0\rangle = 0\rangle, Z 1\rangle = - 1\rangle$	
Hadamard	H	$\frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$	$H 0\rangle = x\rangle_+, H 1\rangle = x\rangle_-$	
T gate	T	$\begin{pmatrix} 1 & 0 \\ 0 & \exp(i\pi/4) \end{pmatrix}$	$T 0\rangle = 0\rangle, T 1\rangle = \exp(i\pi/4) 1\rangle$	
Phase gate	P or S	$\begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$	$S 0\rangle = 0\rangle, S 1\rangle = i 1\rangle$	

- the qubit $|1\rangle$ is at the south pole if $\alpha = \pi$ and $\beta = 0$.

2. *qubit entanglement* describes the state of qubits that are entangled with each other. If the quantum state of two qubits cannot be expressed as a product state, such as $|\psi\rangle_{AB} = |\psi\rangle_A |\psi\rangle_B$, then the state is called an entangled state. Additionally, suppose we know the quantum state of the first qubit with some probability less than one. In that case, we can determine the quantum state of the second qubit with certainty, even if the entangled qubits are located in different places.

To illustrate, consider a quantum state from the book [8]:

$$|\psi\rangle_{AB} = \cos\theta |0\rangle_A |0\rangle_B + \sin\theta |1\rangle_A |1\rangle_B, \quad (2.10)$$

where $|0\rangle_A |0\rangle_B = |0\rangle_A \otimes |0\rangle_B$, $|1\rangle_A |1\rangle_B = |1\rangle_A \otimes |1\rangle_B$, and \otimes is a tensor product. If $\theta = \pi$ then a two-qubit state $|\psi\rangle_{AB}$ can be written as a product state $|\psi\rangle_{AB} = |\psi\rangle_A |\psi\rangle_B$ where $|\psi\rangle_A = |1\rangle_A$, $|\psi\rangle_B = |1\rangle_B$, and A and B states are not an entangled state. If $\theta = \pi/4$, then the two-qubit state cannot be written as a product state $|\psi\rangle_{AB} \neq |\psi\rangle_A |\psi\rangle_B$, and A and B states become an entangled state called the Bell state $|\psi\rangle_{AB} = 1/\sqrt{2}(|0\rangle_A |0\rangle_B + |1\rangle_A |1\rangle_B)$.

3. *parameterized and non-parameterized quantum gates* are used to encode data points in input qubits and to create interactions and entanglement among qubits, which makes them more powerful than instructions on a conventional computer.

A set of unitary quantum gates can operate as parameterized quantum gates [62]:

$$U = U(\gamma, \theta, \phi) = U_\gamma U_\theta U_\phi, \quad (2.11)$$

where γ , θ , and ϕ are parameters (angles in radians), and U in matrix form:

Table 2.2: Some multi-qubit quantum gates. Here, \oplus denotes the XOR operation, and $i, j, k \in \{0, 1\}$.

Name	Matrix	Operation	Gate Diagram
CNOT	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$	$CNOT i\rangle j\rangle = i\rangle i \oplus j\rangle$,	
SWAP	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$	$SWAP i\rangle j\rangle = j\rangle i\rangle$	
CCNOT	$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$	$CCNOT ijk\rangle = i\rangle j\rangle k \oplus (ij)\rangle$	

$$U_\gamma = \begin{pmatrix} e^{i\gamma} & 0 \\ 0 & e^{-i\gamma} \end{pmatrix}, \quad U_\theta = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}, \quad U_\phi = \begin{pmatrix} e^{i\phi} & 0 \\ 0 & e^{-i\phi} \end{pmatrix}. \quad (2.12)$$

A family of quantum gate sets that do not require parameters exist, including single-qubit and multi-qubit gates. Some examples of single-qubit gates as illustrated in Table 2.1 include Pauli gates,

$$I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad (2.13)$$

the Hadamard gate H , the T gate, and the phase gate P ,

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \quad T = \begin{pmatrix} 1 & 0 \\ 0 & \exp(i\pi/4) \end{pmatrix}, \quad P = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}. \quad (2.14)$$

Quantum gates map a qubit to another qubit while preserving the norm of amplitudes. For example, applying the X gate to a qubit in the state $|0\rangle$ results in the state $|1\rangle$, and applying it to a qubit in the state $|1\rangle$ results in the state $|0\rangle$. Similarly, applying the X gate to a superposition state $|\psi\rangle_0 = a_0|0\rangle + b_0|1\rangle$ results in a new state $|\psi\rangle_1 = a_0|1\rangle + b_0|0\rangle$, where $|a_0|^2 + |b_0|^2 = 1$. For more information on single- and multi-qubit gates, please visit [W6].

Multi-qubit gates include two-qubit controlled-NOT (CNOT) gates, two-qubit SWAP gates, and three-qubit controlled-controlled-NOT (CCNOT) gates, sometimes interchangeable with the Toffoli gate [62]. The CNOT gate acts on the second qubit only if the first qubit is in the state $|1\rangle$, while the SWAP gate swaps the states of the two

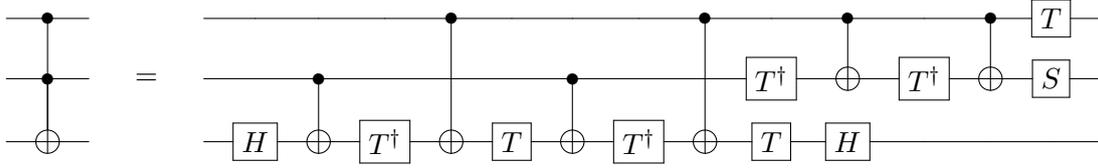


Figure 2.5: Decomposed a Toffoli gate into a universal quantum gate set.

qubits. The CCNOT gate acts on the third qubit only if the first two qubits are in the state $|1\rangle|1\rangle$ (shown in Table 2.2). For further information on single- and multi-qubit gates, please refer to [W6].

4. *measurement* is performed on the qubits based on the set of observables: $O \in \{X, Y, Z, I\}$ after manipulating the input qubits using parameterized and non-parameterized quantum gates. The probability of obtaining a quantum state $M = |i\rangle\langle i|$, $i \in \{0, 1\}$ and the expectation value of an observable O can be calculated using U gates for encoding data points in the qubits and V gates representing single- and multi-qubit gates. The equations for the probability and expectation value are as follows:

$$P(i) = \langle \psi | U^\dagger V^\dagger M V U | \psi \rangle, \quad (2.15)$$

$$\langle O \rangle = \langle \psi | U^\dagger V^\dagger O V U | \psi \rangle. \quad (2.16)$$

2.2.1 Universal Quantum Computation

In conventional classical computing, a function on bits can be calculated using classical logic gates such as OR, AND, and XOR. Any classical logic gate can be constructed using a universal classical logic gate sequence. In classical computing, a NAND logic gate is a universal logic gate. In quantum computing, the CCNOT quantum gates, which are known as Toffoli gates, can mimic classical NAND logic gates for computing a function [62]. Therefore, any function on bits computed on a classical computer can also be executed on a quantum computer.

In universal quantum computation, the Toffoli gates can be approximately expressed using a universal quantum gate set, which is $\{\text{CNOT}, \text{H}, \text{S}, \text{T}\}$, known as Clifford+T. The CNOT, H, and S gates are known as Clifford quantum gates, while the T gate is called a non-Clifford quantum gate. The Gottesman-Knill theorem states that a classical computer can efficiently simulate Clifford gates [35]. However, it cannot simulate non-Clifford quantum gates in a reasonable polynomial time on a classical computer. A quantum computer having around $\mathcal{O}(10^{15})$ non-Clifford quantum gates cannot be simulated efficiently on a classical computer and surpasses the performance of a conventional classical computer [83]. Note that sets of universal quantum gate sets exist depending on a particular quantum error correction technique, such as a qubit flip error.

The universal quantum gate set $\{\text{CNOT}, \text{H}, \text{S}, \text{T}\}$ is extensively used among universal quantum gate sets because it can be implemented efficiently on a promising quantum error correction technique called a *surface code* for building a fault-tolerant digital quantum computer [33]. The universal quantum gate set $\{\text{CNOT}, \text{H}, \text{S}, \text{T}\}$, e.g., illustrated for the Toffoli gate in Figure 2.5, can express and mimic any quantum gate. Moreover, a quantum gate error rate is around 10^{-3} , while a conventional transistor error rate is around 10^{-17} (Chapter 10 of [62]). Thus, correcting quantum errors at a sufficiently high rate is inevitable.

The quantum errors are the bottleneck for operating practical large-scale digital quantum computers surpassing the performance of supercomputers [4].

2.3 Quantum Artificial Intelligence

QAI (or QML) studies quantum learning models for processing datasets faster and better than some conventional AI models. It can be seen as the synergy between AI and QC or AI models boosted by QC. Several quantum learning models theoretically surpass their classical counterparts, such as Quantum Principal Component Analysis (QPCA) [54], a Quantum Support Vector Machine (QSVM) [81, 102], an HHL algorithm for least squares fitting [39] and a QNN, the quantum version of a conventional CNN [2, 11]. The quantum advantages of some QML approaches have been also experimentally demonstrated in laboratories [85, 103, 106]. However, there are no demonstrated quantum advantages of QML approaches for real-world problems on NISQ computers over conventional AI models. Furthermore, QML approaches must know the noise in qubits and quantum gates.

VQAs are QML approaches promising computational advantage in the presence of noise in a quantum computer over conventional classical ML algorithms [21, 31, 40]. Here, we present the general architectures of VQAs for NISQ computers [10, 59] as we do not know when a perfect quantum computer having more than $n \geq 100$ error-free qubits and quantum gates will be available [22, 78, 92].

2.3.1 Variational Quantum Algorithms

VQAs are a QAI approach designed specifically for a faulty digital quantum computer with the help of conventional classical computer. They process input data points similar to a traditional ANN and CNN presented in Chapter 2.1. However, the quality of VQAs relies heavily on embedding classical data in input qubits [89, 90, 109], and on the depth of their parameterized layer for supervised learning tasks [21, 104]. The VQAs comprise three main layers illustrated in Figure 2.6: 1. a *feature-embedding* quantum layer, 2. a *PQC* layer, and 3. a *data read-out* layer.

1. a feature-embedding quantum layer

When dealing with classical datasets $\mathcal{S} = \{\mathbf{x}_j, y_j\}_{j=1}^N$, it is necessary to embed each classical data point $\mathbf{x}_j \in \mathbb{R}^{m \times m}$ in input qubits. We use a feature-embedding quantum layer denoted by \mathcal{X} , which maps the classical data point onto qubits. Commonly used feature-embedding methods include [109]:

1. *basis encoding* is a method that involves mapping a classical data point, denoted as \mathbf{x}_j , to a quantum basis state $|i\rangle$, where i ranges from 0 to $2^n - 1$, and is defined in Equation (2.8). This is achieved through the use of a quantum embedding layer \mathcal{X} , such that the mapping can be expressed as follows:

$$\mathcal{X} : \mathbf{x}_j \rightarrow |i_2\rangle, \quad (2.17)$$

where $|i_2\rangle$ is the bitstring representation of $|i\rangle$. For instance, let us consider the classical data point $\mathbf{x}_1 = (0, 1, 0)^T$. The corresponding quantum basis state is $|i_2\rangle = |010\rangle$, which is obtained using the feature-embedding quantum layer $\mathcal{X} = I \otimes X \otimes I$, acting on the input qubits $|000\rangle$.

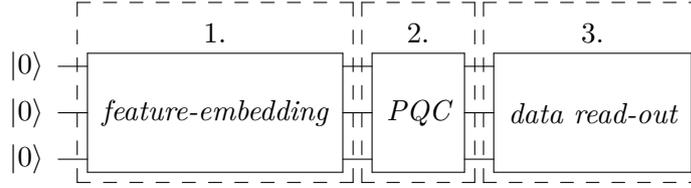


Figure 2.6: The visual representation for VQAs: 1. a *feature-embedding* quantum layer embeds classical data points in qubits, 2. a *PQC* layer manipulates the initialized qubits for supervised learning tasks such as regression and classification, and 3. a *data read-out* layer that generates a solution to the computational task.

2. *amplitude encoding* is a method to map a classical data point \mathbf{x}_j into the amplitude c_i of a quantum state. This quantum state is expressed in Equation (2.8). The mapping is made using the function \mathcal{X} , which takes the classical data point \mathbf{x}_j as input and returns the quantum state $|\psi\rangle$. The quantum state is defined as follows:

$$|\psi\rangle = \sum_{i=0}^{m-1} x_i |i\rangle \quad \text{such that} \quad \sum_i |x_i|^2 = 1, \quad (2.18)$$

where $m = 2^n$, and x_i is an element of the data point \mathbf{x}_j . To illustrate this, consider an example data point $\mathbf{x}_1 = (0.86, 0.50)^T$. The corresponding quantum state is $|\psi\rangle = 0.86|0\rangle + 0.50|1\rangle$. This state is obtained by applying the feature-embedding quantum layer $\mathcal{X} = U_{\theta=\pi/6}$, as defined in Equation (2.12), to the input qubit $|0\rangle$.

3. *angle encoding* is a method to map a classical data point \mathbf{x}_j to a quantum state $|\psi\rangle$. This is done using a feature-embedding quantum layer \mathcal{X} , which consists of a list of U gates that act on the input qubit. Specifically, the input qubit is transformed via the following equation:

$$U(x_0, x_1, x_2) |0\rangle. \quad (2.19)$$

For example, if we have the data point $\mathbf{x}_1 = (0.00, 0.00, 1.57)^T$, then we can apply the gate $U(0.00, 0.00, 1.57)$ to the input qubit $|0\rangle$ to obtain the quantum state $|\psi\rangle$.

2. a parameterized quantum circuit layer

After a feature-embedding quantum layer embedding data points in input qubits, a Parameterized Quantum Circuit (PQC) can be designed using parameterized and non-parameterized quantum gates [95]. Here, the trainable weights are represented by the parameterized quantum gates, which are combined with non-parameterized gates, to provide the PQC with its quantum advantage over AI models. Additionally, an L -layer PQC can be created by repeating the PQC L times, similar to ANNs and CNNs having L hidden layers illustrated in Figure 2.1. An L -layer PQC is interchangeable with an L -depth PQC, which can be mathematically expressed as:

$$V_L(\mathbf{w}) = \prod_{l=1}^L \mathcal{W}_l \cdot \mathcal{V}_l(\mathbf{w}_l), \quad (2.20)$$

where \mathcal{W}_l is a non-parameterized gate for creating entanglement between the initialized qubits using a feature-embedding quantum layer, and

$$\mathcal{V}_l(\mathbf{w}_l) = \prod_k \mathcal{V}_l(\mathbf{w}_l^k) \quad (2.21)$$

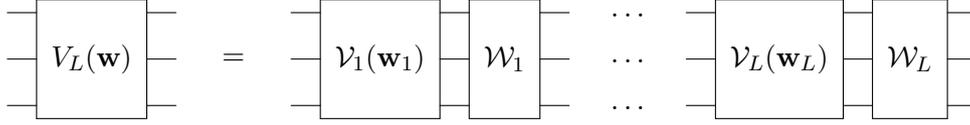


Figure 2.7: A gate diagram representation for the L -depth PQC denoted by $V_L(\mathbf{w})$ having, e.g., three wires/initialized qubits, $k = 0, 1, 2$.

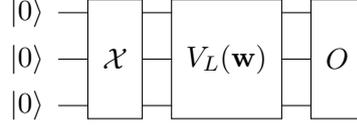


Figure 2.8: A gate diagram representing the feature-embedding, parameterized quantum circuit (PQC), and data read-out layers for a quantum system with three input qubits.

is a parameterized gate, where k represents a number of wires or an input qubit illustrated in Figure 2.7.

3. a data read-out layer

Selecting a feature-embedding method for encoding a problem in the input qubits and designing a PQC depend on the computational task. The objective is to generate a much better result than for its classical counterpart. After the PQC layer generates a result state, we read it for an observable, such as $O \in \{X, Y, Z, I\}$. This reading process is called a measurement. Further, we use the equation below to calculate the observable based on Figure 2.8:

$$\langle O \rangle = \langle 000 \dots 0 | \mathcal{X}^\dagger V_L^\dagger(\mathbf{w}) O V_L(\mathbf{w}) \mathcal{X} | 000 \dots 0 \rangle. \quad (2.22)$$

For a classification task, in order to learn the weight \mathbf{w} , we minimize the loss function:

$$\mathcal{L}_{\mathbf{w}}(\tilde{y}_j, y_j) = 1 - y_j \langle 0 | \mathcal{X}^\dagger V_L^\dagger(\mathbf{w}) O V_L(\mathbf{w}) \mathcal{X} | 0 \rangle \quad \text{if } y_j \in \{-1, +1\}, \quad (2.23)$$

where $\tilde{y}_j = \langle O \rangle = \langle 0 | \mathcal{X}^\dagger V_L^\dagger(\mathbf{w}) O V_L(\mathbf{w}) \mathcal{X} | 0 \rangle \in [-1, +1]$, and $O \in \{X, Y, Z, I\}$. The optimization process is carried out using the following equation which is identical to the one used in training DNNs as presented in Equation (2.6), Chapter 2.1.1:

$$\mathbf{w}_{new} = \mathbf{w}_{old} - \lambda \frac{\partial \mathcal{L}_{\mathbf{w}}(\tilde{y}_j, y_j)}{\partial \mathbf{w}}, \quad (2.24)$$

here, λ represents the learning rate. It is important to note that we minimize the loss function on a classical computer using a classical optimizer since the expectation value $\langle O \rangle$ is a function. For a detailed discussion on calculating a gradient of parameterized quantum gates without the help of a classical computer, refer to [58, 109]. Additionally, the reader is referred to [45, 49] for training VQAs for unsupervised learning tasks.

2.4 Quantum Annealing

A quantum annealing algorithm uses quantum tunneling and entanglement to find solutions to some hard computational problems. It is a meta-heuristic algorithm based on adiabatic quantum computation [5, 30, 36]. This differs from classical simulated annealing, which uses

thermal fluctuation to tackle a specific kind of computational problems [14]. A quantum annealing device, such as the one manufactured by D-Wave Systems, can only determine solutions to combinatorial optimization problems, known as QUBO problems [41, 56]. Its performance depends on setting some hyperparameters, such as an annealing schedule and the qubit connectivity. The device is used in various applications, from aeronautics [96] to quantum chemistry [7] and machine learning [25, 64]. However, due to the difficulty of setting its hyperparameters, a programmable quantum annealing device has yet to demonstrate a quantum advantage over a classical computer for solving real-world combinatorial optimization problems.

In Chapter 3, we present and explain how to map real-world optimization problems to QUBO problems and deploy them on a D-Wave quantum annealer. We also discuss how it can be used in ML techniques such as QSVM and optimization problems [65, 102].

2.4.1 Quantum Annealing Device

The solution generated by quantum annealing devices is similar to finding the most probable configuration of its qubit states s :

$$\tilde{\mathbf{s}} = \arg \max_{\mathbf{s}} p(\mathbf{s}), \quad p(\mathbf{s}) \approx e^{-\beta H(\mathbf{s})}, \quad \mathcal{H}(\sigma_z) |0\rangle = H(\mathbf{s}) |0\rangle, \quad (2.25)$$

Here, β is a scaling parameter, $\mathcal{H}(\sigma_z)$ is the Ising model, $\mathcal{H}(\mathbf{s})$ is its eigenvalue, J_{ij} is the interaction strength of the nearest neighbor spins $\langle \sigma_z^i, \sigma_z^j \rangle$, σ_z^i is a Pauli-z operator having an eigenvalue s_i , and h_i is its bias. The Ising model, denoted by $\mathcal{H}(\sigma_z)$, is given by:

$$\begin{aligned} \mathcal{H}(\sigma_z) &= - \sum_{i < j} J_{ij} \sigma_z^i \sigma_z^j - \sum_i h_i \sigma_z^i, \\ \mathcal{H}(\mathbf{s}) &= - \sum_{i < j} J_{ij} s_i s_j - \sum_i h_i s_i, \quad s_i \in \{-1, +1\}; \end{aligned} \quad (2.26)$$

On the other hand, Equation (2.25) is equivalent to the optimization problem:

$$\tilde{\mathbf{s}} = \arg \min_{\mathbf{s}} \mathcal{H}(\mathbf{s}), \quad \mathbf{s} \in \{-1, +1\}^*. \quad (2.27)$$

By inserting $s_i = 1 - 2b_i$ in $\mathcal{H}(\mathbf{s})$, we obtain its so-called QUBO form [10]:

$$\mathcal{H}(\mathbf{b}) = \sum_{i < j} q_{ij} b_i b_j + \sum_i q_i b_i + \text{const} = \mathbf{b}^T \mathbf{Q} \mathbf{b}, \quad b_i \in \{0, +1\}, \quad (2.28)$$

where q_{ij} is the interaction strength between the spins $\langle b_i, b_j \rangle$, and q_i is its bias.

$$\tilde{\mathbf{b}} = \arg \min_{\mathbf{b}} \mathcal{H}(\mathbf{b}), \quad \mathbf{b} \in \{0, +1\}^* \quad (2.29)$$

In particular, the Ising model has variables $+1$ and -1 , while the QUBO model has variables 0 and $+1$.

2.5 Earth Observation

Satellite images for Earth observation tasks are acquired using airborne or satellite platforms to combat environmental and climate challenges and inform people about potential environmental hazards [63, 77]. The acquired images are available in digital format and contain three pieces of information [W7]:

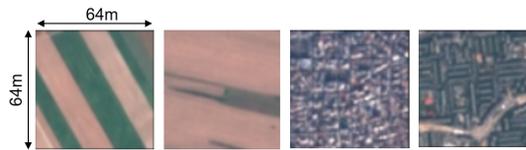


Figure 2.9: Example images of the Eurosat dataset with two labels and RGB (red-green-blue) spectral bands: “annual crop” and “residential area.”



Figure 2.10: Example images of the UC Merced Land Use dataset with two labels and RGB (red-green-blue) spectral bands.

- *spectral bands* refer to sensed signals with different wavelengths,
- *radiometric resolution* is the number of radiometry levels of remotely sensed signals,
- *pixel spacing* is the on-ground pixel spacing of the remotely sensed images.

Satellite image datasets provided by organizations like the European Space Agency (ESA) are massive compared with conventional digital red-green-blue (RGB) images [82]. For example, a single Sentinel-2 image frame of $200 \times 200 \text{ km}^2$ having 12 spectral bands, eight bits of radiometric resolution per spatial resolution, and a pixel spacing of 60 m has around 11 million heterogeneous pixels and carries information of $1.067 \cdot 10^9$ bits, requiring 1.067 Gbit memory in the storage device [84]. The information quantity even increases dramatically with data collected day and night by satellites, such as 1.6 TB of compressed images per day collected by a single Sentinel-2 satellite. This has made benchmark datasets widely available for AI practitioners [23].

2.5.1 Benchmark Satellite Datasets

Benchmark satellite datasets assist in analyzing the performance of emerging learning models, such as QML, compared to the best existing models. They differ in their characteristic, heterogeneity, quality, and volume. This section introduces diverse benchmark datasets this thesis used.

An Eurosat dataset: the dataset in question is a collection of Sentinel-2 SAR images featuring 27,000 images that have been labeled (i.e., coarsely classified) and geo-referenced. This dataset is called Eurosat and comprises a large number of patch-based images, each measuring 64×64 pixels and belonging to one of ten distinct classes. Each image has 13 spectral bands ranging from 443 nm to 2,190 nm, with a spatial pixel spacing of 10 m/pixel to 60 m/pixel. You can look at Figure 2.9 for a visual representation of this dataset [43].

A UC Merced dataset: it comprises 2,100 image scenes with three RGB spectral bands. Each image has a spatial resolution of 256×256 pixels and covers 21 classes. The dataset includes a variety of classes, ranging from the smallest class, such as “tennis court,” which

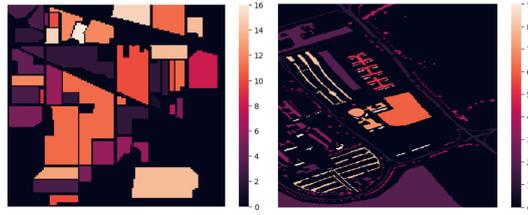


Figure 2.11: [Left] The hyperspectral image of Indian Pine, USA and [Right] the hyperspectral image of Pavia University, Italy.

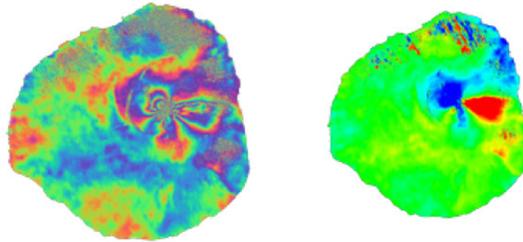


Figure 2.12: Interferometric Synthetic Aperture Radar (**InSAR**) images of the Cape Verde volcano for monitoring its deformation: [Left] the phase change of two **SAR** images of Cape Verde volcano. The two **SAR** images are obtained at different times and positions. [Right] its deformation is shown in red [26].

has 42 data points, to the largest classes, like “agricultural,” which has 100 data points. You can see an example of the dataset in Figure 2.10.

The HSI dataset of Indian Pine and Pavia University: an Indian Pine **HSI** is obtained using the Imaging Spectroscopy and the Airborne Visible/Infrared Imaging Spectrometer (**AVIRIS**) sensor mounted on an airborne platform. It contains 220 contiguous spectral bands ranging from 400 nm to 2,500 nm at 10 nm intervals and generates 16 classes [W2]. Its spatial dimension is 240×240 pixels. On the other hand, the Pavia University **HSI** is a 3rd order tensor $\mathcal{S} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ with a spatial dimension of 610×340 pixels and 9 classes. 103 contiguous spectral bands characterize each pixel shown in Figure 2.11. It is important to note that the number of classes may vary depending on the interpretation task.

An InSAR dataset: an imaging device called **SAR** obtains complex-valued images by utilizing coherent signals. In contrast, **InSAR** is an inversion technique that monitors deformations on the Earth’s surface by analyzing and comparing the phase values of two or more complex-valued images captured by **SAR** devices [76]. You can see an illustration of the **InSAR** image in Figure 2.12.

PolSAR datasets: **PolSAR** datasets are unique compared with other remotely-sensed datasets because they provide physical scattering information of ground targets. **PolSAR** images are represented by a pixel scattering scheme based on the physical scattering information of ground targets illustrated in Figure 2.13, and their pixels are represented by 2×2 scattering matrices [38]:

$$S = \begin{pmatrix} s_{hh} & s_{hv} \\ s_{vh} & s_{vv} \end{pmatrix}, \quad (2.30)$$

where each element of S is a complex-valued number, with the first index representing the polarization state of the incident polarized beam and the second index representing the polarization state of the reflected polarized beam on targets. By assuming incident polarized states (i.e., incident Jones vectors \mathbf{J}_i), we can represent **PolSAR** images by their reflected

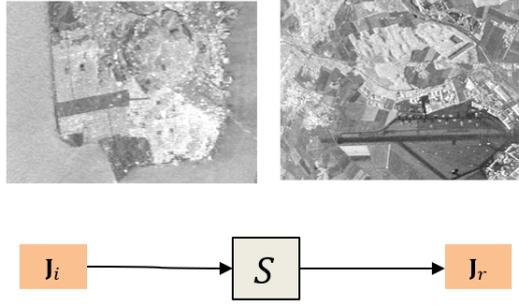


Figure 2.13: Here, we present a C-band PolSAR image (with the wavelength range from 1,530 nm to 1,565 nm) of California, USA obtained from Radarsat-2, and an L-band PolSAR image (with the wavelength range from 1,565 nm to 1,625 nm) of Oberpfaffenhofen, Germany taken by the airborne ESAR-L instrument. [Top] PolSAR images of California and Oberpfaffenhofen, [Bottom] an incident Jones vector \mathbf{J}_i , the polarization-changing targets in the PolSAR image S , and a reflected Jones vector \mathbf{J}_r .

polarized states (i.e., reflected Jones vectors \mathbf{J}_r):

$$\mathbf{J}_i \xrightarrow{S} \mathbf{J}_r : S\mathbf{J}_i = \mathbf{J}_r. \quad (2.31)$$

2.6 Brief Summary for Applications of Faulty Quantum Computers

This chapter provides brief overviews of our key findings related to implementing QML on faulty quantum computers using benchmark satellite datasets. The next chapter will delve into more detailed steps for implementing hybrid classical-quantum and quantum annealing approaches on faulty quantum computers. Chapters 4 and 5 present our main results based on several articles.

We investigate and study the design and applications of hybrid classical-quantum approaches using benchmark multispectral and polarimetric satellite datasets. More detailed discussions can be found in the publications referenced in Chapter 4.1 of this dissertation.

For multispectral images, we propose a two-level encoding approach. The first level uses a deep neural network to reduce the spatial dimensionality of the multispectral images so that the outputs d of the deep neural networks are compatible with the input qubits $n \geq d$ [94]. The second level employs the basis encoding to embed the deep neural network outputs in the input qubits. With this encoding technique, we design and analyze a PQC layer based on an energy-based model. We minimize its loss function using a gradient descent expressed in Equation (2.24) on a classical computer. We also provide a quantum transfer learning model for benchmarking QML approaches on classical image scene datasets, the quantum version of classical transfer learning. We conduct our first experiments on a digital quantum simulator with 17 input qubits [16], and show that our hybrid classical-quantum model produced better classification outcomes than a classical deep neural network.

For polarimetric images, we characterize them using their Stokes parameters, which are the doppelgänger of qubits. We encode them in qubits using the angle encoding without the classical dimensionality-reduction layer due to the doppelgänger feature, which results in a one-to-one mapping between the Stokes parameters and input qubits. We conduct our experiments on the IBM quantum computer with five input qubits [W8]. We name the one-to-one mapping of the Stokes parameters to the input qubits a *natural embedding*. We also demonstrate that the one-to-one mapping based on the underlying physics of polarimetric

images conveys physical information. More importantly, the one-to-one mapping technique overcomes the out-of-distribution problem, typical of [AI](#). Our methodology for polarimetric datasets can be used to benchmark future hybrid classical-quantum models.

We implement also quadratic programming problems on a [QA](#) device. This is achieved by designing and analyzing a novel mechanism that transforms quadratic programming problems into [QUBO](#) problems. The quadratic programming problems in this thesis include a [SVM](#), a network flow minimization problem, and a quantum boosting algorithm. More importantly, we devise strategies to map any hard quadratic programming problem to its [QUBO](#) form and execute it on a quantum annealer by following our workflow defined in [Chapter 3.2.1](#). In some pattern recognition tasks, the outputs of the quantum annealer generated distinct classes better than conventional learning methods and a classical annealer. For more detailed discussions, the reader is referred to the publications in [Chapter 4.2](#).

To the best of our knowledge, these studies are the first attempt to design a methodology for embedding satellite datasets in faulty quantum computers and to explore and study potential [EO](#) applications for hybrid classical-quantum approaches. This assessment will help design experiments for current and future quantum computers, by considering their potential benefits and imperfections, such as a coherent Ising machine [\[20\]](#) or a digital annealer provided by e.g., Qilimanjaro Tech [\[W9\]](#).

3

Methodology

Chapter 2 of this dissertation provided background information on VQAs and QA, highlighting the synergy between AI and QC. We also presented the brief summary on our findings when processing hybrid classical-quantum approaches on benchmark satellite datasets.

A quantum computer, which utilizes superposition and entanglement of qubits, has a computational advantage for finding solutions to some challenging computational problems, including AI models [11] and integer optimization problems [29]. The computational power of a quantum computer is based on the fact that a sufficiently high number of entangled qubits is classically non-simulatable, making it an inspiring area of research for tackling some practical EO problems [35].

There are challenges in encoding satellite datasets into qubits and in understanding the potential advantages and limitations of QML models compared to traditional AI ones [55, 66, 67, 109]. Additionally, quantum computers on the current market are limited by the number of error-prone qubits and quantum gates, which restricts the size of QML models [W10, W8].

This chapter will explain the techniques to embed data points in small faulty quantum computers when considering satellite datasets and EO methodologies. We also demonstrate how to deploy hybrid classical-quantum and quantum annealing approaches on NISQ computers and QA devices, respectively. Finally, we discuss the prospects and bottlenecks of hybrid classical-quantum and quantum annealing approaches.

3.1 Hybrid Classical-Quantum Approach

Real-world datasets from sensors mounted on satellites and airplanes have unique features and dimensions. For example, multispectral satellite images mostly have three or more spectral bands and spatial information $\mathbf{x}_j \in \mathbb{R}^{m \times m}$. In comparison, hyperspectral satellite images have up to about 200 spectral bands and a spatial dimensionality $\mathbf{x}_j \in \mathbb{R}^{m \times m}$ [W2, 23, 43]. However, executing VQAs on high-dimensional satellite datasets on NISQ computers for supervised learning tasks can be challenging due to the limited number of available qubits. Typically, $n < m \times m$, where n is the number of qubits, and NISQ computers today have around $n \leq 100$ qubits. This means we must embed the data point in the n number of qubits, being much smaller than the total number of elements. To address this bottleneck, this thesis proposes a *two-level encoding* technique for embedding classical datasets in small imperfect qubits [66]:

- the *first-level encoding* performs either a classical dimensionality reduction or feature selection technique on large-scale high-dimensional satellite datasets, and
- the *second-level encoding* (or a feature-embedding quantum layer) then embeds the output of the first-level encoding in the input qubits by leveraging a feature-embedding quantum method presented in Chapter 2.3.1.

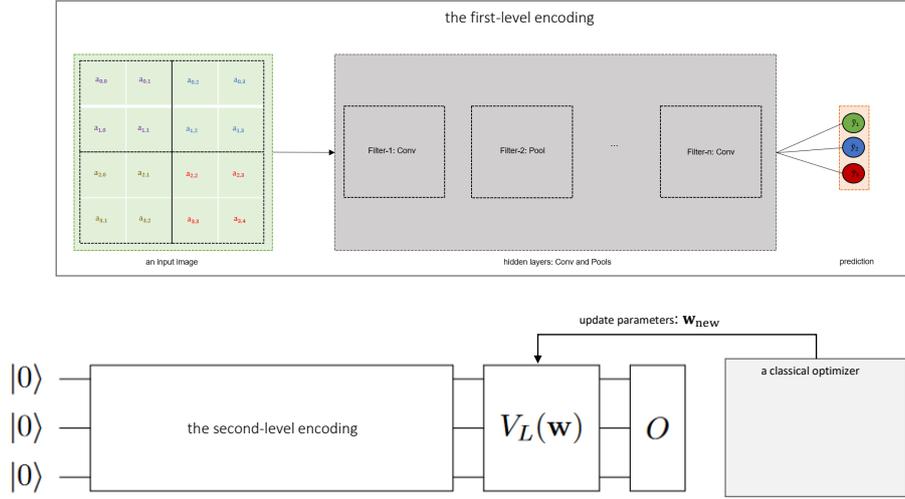


Figure 3.1: An example hybrid classical-quantum approach having three input qubits for a high-dimensional satellite image $\mathbf{x}_j = (a_{0,0}, a_{0,1}, \dots, a_{3,4})^T \in \mathbb{R}^{4 \times 4}$. Here, the first classical layer is optional, depending on given features of classical datasets.

Once the data points are embedded in the qubits via a two-level encoding technique, we optimize the adjustable weights of the PQC denoted by $V_L(\mathbf{w})$ using a classical optimizer. This approach to training VQAs on satellite datasets with the help of the two-level encoding and classical optimizer is a hybrid classical-quantum approach illustrated in Figure 3.1 [10, 16].

3.1.1 Workflow Details for Hybrid Classical-Quantum Approaches

As seen in the prior sections, hybrid classical-quantum approaches combine the strengths of quantum computing and classical computing. We can enhance conventional classical computational science by integrating quantum computing resources into traditional hardware and algorithms. In this context, we present the workflow details for hybrid classical-quantum approaches:

1. Firstly, we represent the images $\mathbf{x}_j \in \mathbb{R}^{m \times m}$ by d elements using traditional dimensionality reduction or feature selection techniques via a *classical layer*. This ensures that $n \geq d$, where the d elements are the most informative and relevant features of a given high-dimensional satellite image.
2. Secondly, using a feature-embedding quantum method in a *quantum layer*, we encode the most informative d elements in the input qubits.
3. Lastly, we define a loss function depending on the supervised learning task, such as regression or classification and optimize it using a conventional *classical optimizer*.

In particular, hybrid classical-quantum approaches for supervised learning tasks comprise: a *classical layer* that reduces the dimensionality or implements feature selection on input images, a *quantum layer* that performs second-level encoding via the feature-embedding quantum method on the input qubits, followed by the PQC and data read-out layers (see VQAs presented in Chapter 2.3.1), and a *classical optimizer* that optimizes the PQC with the help of traditional classical computing resources.

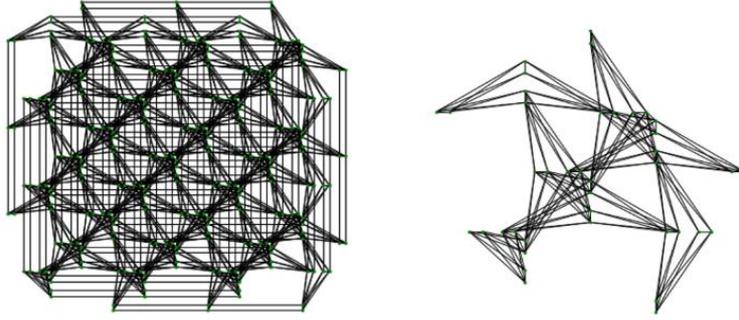


Figure 3.2: The Pegasus topology of the D-Wave quantum annealer: green dots are spins, and the black lines represent their interaction J .

3.1.2 Prospects and Bottlenecks

There are several extensions to the research in this thesis. The experiments can take several hours to several days when training hybrid classical-quantum models on real-world big datasets using real NISQ computers [67]. In contrast, classical computers can produce results in just a few seconds. Despite this, NISQ computers may output better results than classical methods [46]. We also must develop efficient strategies and novel methods for training quantum machine learning approaches with a complete error-mitigation and -correction code to achieve quantum advantage using future quantum computers. To obtain the so-called quantum advantage, an efficient method for encoding classical datasets in qubits is also required [109]. Here, the input dataset plays a vital role in quantum algorithms. Moreover, we should find more practically relevant problems to utilize and benchmark distinct quantum computers, such as superconducting, photonic-based, or neutral atom-based concepts.

3.2 Quantum Annealing Approach

A quantum annealing approach refers to processing quadratic programming problems on a programmable quantum annealing device. An example is a D-Wave quantum annealer similar to an analog quantum simulator [37]. It comprises spins arranged according to the so-called Pegasus topology $G = (v, e)$. The spins are located on the vertices v , and the edges e define the interacting spins. However, the Pegasus topology does not allow for all-to-all connectivity between spins, as illustrated in Figure 3.2. The currently available D-Wave quantum annealer has also around 5,640 spins. As mentioned earlier in Chapter 2 of this dissertation, it can only tackle a combinatorial optimization problem called an Ising model, equivalent to a QUBO problem [W10, 68, 102].

In particular, a quantum annealer promises to reach better local minima to a QUBO problem than a classical annealer by using adiabatic quantum computing defined by:

$$\mathcal{H}(\lambda(t)) = (1 - \lambda(t))\mathcal{H}_0 + \lambda(t)\mathcal{H}(\sigma_z), \quad H_0 = - \sum_i \sigma_x^i, \quad (3.1)$$

where $\lambda(t) \in [0, 1]$, $0 \leq t \leq \tau$, and \mathcal{H}_0 is the initial Hamiltonian. Moreover, adiabatic quantum computing implies that a quantum annealer starts with the Hamiltonian \mathcal{H}_0 and ends with the problem Hamiltonian $\mathcal{H}(\mathbf{s})$ having $\tilde{\mathbf{s}} = \mathbf{s}$ when varying $\lambda(t = 0) = 0$ to $\lambda(t = \tau) = 1$ slowly enough during the annealing time $0 \leq t \leq \tau$ (often given in microseconds) [30, 41]. A way of varying $\lambda(t)$ is called an annealing schedule expressed by $[\lambda(t), t]$.

It is still unknown how slowly one should vary $\lambda(t)$ to gain some quantum advantage for computing real-world problems over the best classical algorithms [12]. In addition, the (logical) variables of any real-world problem need to be embedded in the physical variables, i.e., the spins at vertices v , due to the limited connectivity of the physical variables in the Pegasus topology $G = (v, e)$. This dissertation, based on the journal papers [64, 65, 68] (see Chapter 4.2), utilizes four sequences of the annealing schedule, and the embedding strategy called the “minor-embedding” offered by D-Wave Systems [W10].

3.2.1 Workflow Details for a Quantum Annealing Approach

For demonstration, we consider an SVM posed as a Quadratic Programming (QP) problem. This is a challenging problem under the NP-hard category [6]. Therefore, it justifies the study and utilization of a quantum annealer for solving QP problems of practical importance [65, 68]. The following equation defines our SVM:

$$\begin{aligned} \text{minimize } \mathcal{H}(\mathbf{a}) &= \frac{1}{2} \sum_{i,j} a_i a_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) - \sum_i a_i, \\ \tilde{\mathbf{a}} &= \arg \min_{\mathbf{a}} \mathcal{H}(\mathbf{a}), \\ \text{subject to } & 0 \leq a_i \leq C, \\ \text{and } & \sum_i a_i y_i = 0, \quad a_i \in \mathbb{R}, \end{aligned} \quad (3.2)$$

where C is a regularization parameter, and $k(\cdot, \cdot)$ is the kernel function of the SVM [60]. It is sometimes called a kernel-based SVM, and the point \mathbf{x}_i with $a_i \neq 0$ is called a support vector. Furthermore, the SVM is a quadratic optimization problem and can be employed as a classifier for supervised learning tasks.

Workflow steps for deploying computational problems on a D-Wave quantum annealer:

1. *QP-to-QUBO*: the problem of QP optimization involves both continuous and discrete variables. On the other hand, the QUBO problem is an optimization problem that uses binary variables $b_i \in \{0, +1\}$, and the D-Wave quantum annealer is especially designed for tackling QUBO problems. Therefore, we need to convert a QP problem to a QUBO problem by using, e.g., an one-hot encoding technique, to represent its variable a_i . We can represent it as follows:

$$a_i = \sum_{k=0}^{K-1} B^k b_{K+i+k}, \quad b_{K+i+k} \in \{0, +1\}, \quad (3.3)$$

where K is the number of binary variables (bits), and $B = 2$ is the base.

The example SVM expressed in Equation (3.2) can be transformed to the QUBO form using Equation (3.3) because it is a QP problem:

$$\begin{aligned} \tilde{\mathbf{b}} &= \arg \min_{\mathbf{b}} \mathcal{H}(\mathbf{b}), \\ \mathcal{H}(\mathbf{b}) &= \frac{1}{2} \sum_{ijkl} b_{K+i+k} b_{K+j+l} B^{k+l} y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) - \sum_{ik} B^k b_{K+i+k} + \varepsilon \left(\sum_{ik} B^k b_{K+i+k} y_i \right)^2 \\ &= \sum_{ij} \sum_{kl} b_{K+i+k} Q_{K+i+k, K+j+l} b_{K+j+l}, \end{aligned} \quad (3.4)$$

where

$$Q_{K_{i+k}, K_{j+l}} = \frac{1}{2} B^{k+l} y_i y_j (k(\mathbf{x}_i, \mathbf{x}_j) + \varepsilon) - \delta_{ij} \delta_{kl} B^k. \quad (3.5)$$

Here, ε is a multiplier for encoding the second constraint in Equation (3.2). We see that our SVM is now the QUBO problem optimized over binary variables $\mathbf{b} \in \{0, 1\}^*$ [65, 102].

If we are given a linear integer programming problem expressed by:

$$\text{minimize } \mathcal{H}(\mathbf{a}) = \sum_i (a_i \mathbf{x}_i - y_i), \quad (3.6)$$

then we first change it to a corresponding QP problem by taking its norm $\|\cdot\|_2$ and then transform this QP problem to the QUBO problem by representing a_i by its one-hot encoding form. If the original problem is already in the QUBO form, we directly move to the next step in the workflow, namely “minor-embedding”.

2. *minor-embedding*: as mentioned earlier, D-Wave Systems has developed a quantum adiabatic device that solves QUBO problems by configuring spins (up or down) on the vertices v of its Pegasus topology $G = (v, e)$. The edges e represent pairwise interactions. To map the logical variables $b_{K_{i+k}}$ to the physical qubits located at the vertices shown in Figure 3.2, we use a heuristic algorithm called “minor-embedding” [18]. This algorithm embeds the logical variables $b_{K_{i+k}}$ in the vertices v and ensures that the edges e reflect the interactions q_{ij} between the binary variables $b_{K_{i+k}}$ and $b_{K_{j+k}}$.
3. *programming D-Wave*: this refers to setting the hyperparameters of a D-Wave quantum annealer, specifically the hyperparameters of the *advantage_system4.1* device [W10]. These hyperparameters include the annealing schedule $[\lambda(t), t]$, the annealing time range $t \in [0.5, 2000]$ microseconds, and the reading range $[1, 10000]$. It is important to note that the setting of these hyperparameters is dependent on the specific problem to be solved and will affect the solutions of QUBO problems produced by the D-Wave quantum annealer.

3.2.2 Prospects and Bottlenecks

There are still many unresolved questions and several areas where further research is needed. For example, how to effectively set the annealing parameters and how to embed computational problems efficiently in the limited connectivity of qubits being present in a quantum annealing device need to be clarified further. Additionally, it is essential to benchmark a D-Wave quantum annealer for supervised learning tasks in comparison with future digital quantum annealers [15, W9] and analog quantum simulators, including coherent Ising machines [20].

4

Main Results I: Gaining Insights into Small-Scale Faulty Quantum Computers

In Chapter 1 of this dissertation, we formulated and identified our research questions hindering the training of QML approaches on satellite datasets. Compared with conventional deep learning and machine learning models, designing and training QML approaches is a complex and intricate process. Additionally, with a limited number of qubits of a quantum computer and a wide range of satellite datasets being available, selecting the appropriate dataset, encoding it in small faulty quantum computers, that is, NISQ and QA devices, and comparing the performance of QML approaches with conventional DL models can be challenging. Thus, we already proposed a two-level encoding technique and hybrid classical-quantum models in Chapter 3 of this dissertation to address these issues, whereas Chapter 2 of this dissertation introduced background concepts and different satellite datasets we used.

4.1 A NISQ Computer for Earth Observation

We study and benchmark QML models deployed on NISQ computers with the help of classical computing resources when using EO datasets. We present a technique to embed classical data points in qubits and demonstrate the potential benefits of using classical-quantum models over conventional DL approaches. It may prove relevant even if no quantum advantage is intended. This chapter is based on three articles:

- **A:** S. Otgonbaatar and M. Datcu, “Classification of Remote Sensing Images with Parameterized Quantum Gates,” in *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1-5, 2022, art no. 8020105, doi: 10.1109/LGRS.2021.3108014 [66].
- **B:** S. Otgonbaatar, G. Schwarz, M. Datcu, and D. Kranzlmüller, “Quantum Transfer Learning for Real-World, Small, and High-Dimensional Remotely Sensed Datasets,” in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 16, pp. 9223-9230, 2023, doi: 10.1109/JSTARS.2023.3316306 [72].
- **C:** S. Otgonbaatar and M. Datcu, “Natural Embedding of the Stokes Parameters of Polarimetric Synthetic Aperture Radar Images in a Gate-based Quantum Computer,” in *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1-8, 2022, art no. 4704008, doi: 10.1109/TGRS.2021.3110056. [67].

4.1.1 Publication A: Classification of Remote Sensing Images with Parameterized Quantum Gates

S. Otgonbaatar and M. Datcu

Abstract. We explore and investigate the potential application of **QML** models. To keep the quantum simulator cost and resources – such as training time and the number of input qubits – as low and small as possible, we use a digital quantum simulator with 17 input qubits, e.g. a Google Tensorflow framework. To encode our classical images in the input qubits, we propose and utilize a two-level encoding strategy introduced in Chapter 3 of this dissertation. Our hybrid classical-quantum approach demonstrates that it generates two-label images with better accuracy than a traditional deep neural network when tested on the real-world image of Berlin, Germany. We use two-label images of the Eurosat dataset comprising annual crop and residential area classes for a satellite dataset. The Eurosat dataset is a unique and high-quality dataset with almost zero mislabeled classes compared with other satellite datasets. It is a large-scale, low-dimensional dataset compared with conventional benchmark RGB image datasets. This work is the first attempt to design a training strategy for **QML** models for satellite datasets. Several extensions of this work exist for training hybrid classical-quantum approaches on multi-label Eurosat images and other satellite datasets (e.g., a UC Merced Land Use dataset) with much smaller scale and higher dimensionality than a large-scale and low-dimensional dataset.

Classification of Remote Sensing Images With Parameterized Quantum Gates

Soronzonbold Otgonbaatar¹ and Mihai Datcu², *Fellow, IEEE*

Abstract—This letter studies how to program and assess a parameterized quantum circuit (PQC) for classifying Earth observation (EO) satellite images. In this exploratory study, we assess a PQC for classifying a two-label EO image dataset and compare it with a classic deep learning classifier. We use the PQC with an input space of only 17 quantum bits (qubits) due to the current limitations of quantum technology. As a real-world image for EO, we selected the Eurosat dataset obtained from multispectral Sentinel-2 images as a training dataset and a Sentinel-2 image of Berlin, Germany, as a test image. However, the high dimensionality of our images is incompatible with the PQC input domain of 17 qubits. Hence, we had to reduce the dimensionality of the input images for this two-label case to a vector with 16 elements; the 17th qubit remains reserved for storing label information. We employed a very deep convolutional network with an autoencoder as a technique for the dimensionality reduction of the input image, and we mapped the dimensionally reduced image onto 16 qubits by means of parameter thresholding. Then, we used a PQC to classify the two-label content of the dimensionally reduced Eurosat image dataset. A PQC classifies the Eurosat images with high accuracy as a classic deep learning method (and with even better accuracy in some instances). From our experiment, we derived an enhanced deeper insight into programming future gate-based quantum computers for many practical problems in EO.

Index Terms—Earth observation (EO), parameterized quantum circuit (PQC), quantum machine learning (QML).

I. INTRODUCTION

MACHINE learning (ML) is a framework of methods to extract underlying features and patterns from heterogeneous big data. By learning features or a probability distribution of patterns in the dataset, one can perform tasks, such as prediction, clustering, or data classifications in hitherto unknown domains. Such ML tasks are elements of supervised and unsupervised learning methods. A supervised learning method has to learn about conditional probability information of the dataset given its correct labels, e.g., classification and prediction [1]. The training of a labeled dataset is a classic example of supervised learning and can be done with high validation accuracy [2]. In contrast, an unsupervised

learning method deals with the datasets that do not yet have the label information, e.g., results generated by clustering. Deep learning is a heuristic method, which can be composed of several layers with artificial neurons [3] in which the connections of each layer are parameterized, while the parameters are tuned by using loss functions. Typical examples include Hopfield networks (HNs) and Boltzmann machines (BMs) [4], which are related to an Ising-type model. BMs have artificial neurons with set of values $\{-1, +1\}$, while HNs have a continuous set of values. In practical applications, BMs are often intractable due to their high-dimensional dataset. This difficulty can be overcome by a restricted Boltzmann machine (RBM).

An RBM is composed of two layers named hidden \mathbf{h} and visible \mathbf{v} . The hidden and visible layers form a bipartite graph, which provides connections between the layers but not within the layers $\mathbf{v} \xleftrightarrow{p(h|v)} \mathbf{h}$. The RBM can be extended with additional hidden layers \mathbf{h}_1 , and the RBM, now called a deep RBM, then forms the layers as $\mathbf{v} \xleftrightarrow{p(h|v)} \mathbf{h} \xleftrightarrow{p(h_1|v)} \mathbf{h}_1$. The deep RBM performs better for training the dataset than other deep learning methods [4] and is employed for data encoding, dimensionality reduction, as well as a feature extraction [5].

In addition, noisy intermediate-scale quantum (NISQ) devices are a novel quantum computing technology providing solutions for sampling from hard-to-simulate probability distributions, e.g., (restricted) BMs, solving high-complexity problems, or supporting ML algorithms for the first time [6]–[8]. An NISQ device is termed a PQC when it has parameterized quantum gates. Sometimes, the PQC is interchangeable with quantum machine learning (QML). As for multispectral images in the Earth observation (EO), deep learning networks can handle big data and are very successful methods for scene classification [9]–[13]. More importantly, they are independent of the input size of the multispectral images, and their outputs are continuous values. However, in our case study, the PQC has 16 qubits as input (due to the current limitations of the quantum technology) and a single qubit as output yielding a binary value depending on the binary state of the qubits. Therefore, we had to focus on how to feed the two-label images of EO into 16 qubits and how to program the parameterized quantum gates of the PQC. As a real-world image dataset of EO, we consider a multispectral Eurosat Sentinel-2 image dataset [14] as a training dataset and a real-world Sentinel-2 image of Berlin, Germany, as a test image; each image in this training dataset is a vector with a size of $64 \times 64 \times 3$ elements, and the test image is a vector with a size of $692 \times 633 \times 3$ elements. Therefore, as already mentioned above, there are several challenges when we feed

Manuscript received September 18, 2020; revised February 19, 2021, May 26, 2021, and July 8, 2021; accepted August 24, 2021. Date of publication September 9, 2021; date of current version January 7, 2022. (Corresponding author: Soronzonbold Otgonbaatar.)

Soronzonbold Otgonbaatar is with the German Aerospace Center (DLR) Oberpfaffenhofen, 82234 Weßling, Germany (e-mail: soronzonbold.otgonbaatar@dlr.de).

Mihai Datcu is with the German Aerospace Center (DLR) Oberpfaffenhofen, 82234 Weßling, Germany, and also with the University POLITEHNICA of Bucharest (UPB), 060042 Bucharest, Romania (e-mail: mihai.datcu@dlr.de).

Digital Object Identifier 10.1109/LGRS.2021.3108014

1558-0571 © 2021 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See <https://www.ieee.org/publications/rights/index.html> for more information.

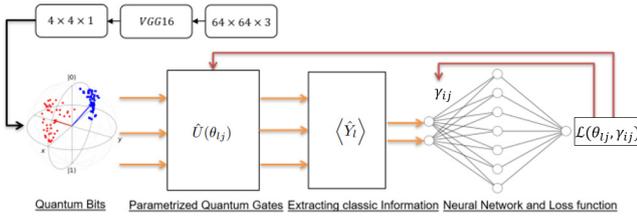


Fig. 1. Hybrid network, a classic and a PQC layer where θ_{ij} and γ_{ij} are training parameters.

these high-dimensional images to the input space of 16 qubits as well as programming the PQC.

To overcome the challenges above mentioned, we introduce a hybrid (classic-quantum) network that is a combination of a conventional computer and the PQC (see Fig. 1). For this hybrid network, we leverage the classic deep learning architecture, a very deep convolutional network (VGG16) [15], and an autoencoder such as RBM to reduce the dimension of the Eurosat image to a vector with a size of $4 \times 4 \times 1$ elements; in particular, the VGG16 network extracts the most meaningful features of the Eurosat images, and the autoencoder encodes these meaningful features to a vector with a size of $4 \times 4 \times 1$ elements. We then feed these encoded features to the inputs of the PQC. Besides, for programming the PQC, we apply a conventional computer to differentiate the loss function and to update the trainable parameters of the quantum gates in the PQC [2]. Thus, this study aims for the leveraging of some classic techniques and a conventional computer to support the PQC and for outlining potential challenges when programming the PQC for a real-world EO dataset.

This letter is structured as follows. In Section II-A, we introduce the dimensionality reduction of the Eurosat image dataset. Then, we devise our parameterized quantum circuit (PQC) for the classification of the dimensionally-reduced images (see Section II-B), and in Section III, we benchmark our PQC for the Eurosat dataset. Finally, we evaluate the developed classification method using the PQC for a remotely sensed image of Berlin, Germany, acquired by Sentinel-2 (see Section IV) and draw some conclusions (see Section V).

II. HYBRID NETWORK

A. Classic Layer for the Dimensionality Reduction of the Eurosat Dataset

We have a hybrid classic-quantum layer, where the classic layer is composed of the VGG16 and the convolutional autoencoder. We use this classic layer for extracting physically meaningful features from the Eurosat image dataset as an RBM. The RBM is capable of encoding the features of the dataset with very few bits. Moreover, the RBMs with binary variables are a special type of HNs and represent an energy-based model in terms of an Ising model. Such a model is used to establish a most probable configuration of particles with binary states, and furthermore, the Ising model can be formulated as

$$E(\mathbf{v}, \mathbf{h}) = - \sum_i b_i h_i - \sum_j c_j v_j - \sum_{i < j} w_{ij} h_i v_j$$

$$P(\mathbf{v}, \mathbf{h}) = \frac{e^{-E(\mathbf{v}, \mathbf{h})}}{\sum_{\mathbf{v}, \mathbf{h}} E(\mathbf{v}, \mathbf{h})}, \quad \mathbf{v}, \mathbf{h} \in \mathbb{R}^n \quad (1)$$

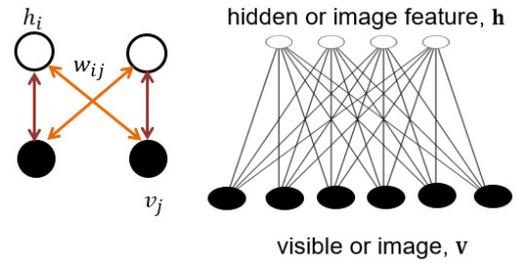


Fig. 2. RBM. An image \mathbf{v} with a size of $64 \times 64 \times 3$ elements is encoded as an another image \mathbf{h} with a size of $4 \times 4 \times 1$.

where b_i, c_j are bias coefficients and w_{ij} represents the strength of interaction between the neighbor variables v_j and h_i , with v_j called a visible variable in a visible layer and h_i called a hidden variable in a hidden layer (see Fig. 2). The probability distribution $P(\mathbf{v}, \mathbf{h})$ defines a most probable configuration of the given variables. This formulation is named after an RBM.

To find the most probable configuration of the visible variable, one needs to integrate out (marginalize) hidden variables and then maximize its log-likelihood distribution. Moreover, we represent an input image as the visible layer, and its selected features are represented in the hidden layer (see Fig. 2). In other words, we can encode an input image by a very small number of the hidden variables in the hidden layer. This encoding procedure is sometimes called dimensionality reduction or autoencoding of the input image.

B. Quantum Layer as a Training Layer

A quantum circuit is an ensemble of quantum gates and quantum bits, and quantum gates (e.i., unitary operation) operate on quantum bits to perform computations (rotations). A collection of quantum gates with trainable parameters is called a PQC. Quantum gates are devised as a unitary operation $\hat{U}(\beta)$, which is expressed as follows:

$$\hat{U}(\beta) = e^{-i\beta\hat{H}} \quad (2)$$

where β is a trainable parameter, \hat{H} represents a collection of 2×2 matrices that are named after Pauli matrices, and the Pauli matrices rotate a quantum bit (qubit) to another quantum bit. Qubits are state vectors, and two state vectors or simply two state qubits are $|0\rangle$ and $|1\rangle$, which are bits in a quantum computer.

We are considering a PQC with 17 input qubits. Hence, we first need to make a dimensionality reduction for the Eurosat images since each image is a vector with the high spatial dimensionality of $\mathbf{v} \in \mathbb{R}^{64 \times 64 \times 3}$, and second, we encode the dimensionally reduced images into qubits $|q\rangle = |q_1\rangle|q_2\rangle, \dots, |q_{16}\rangle, |q_j\rangle \in \{|0\rangle, |1\rangle\}$ named “data qubits,” and their labels $l(\mathbf{v}) \in \{-1, +1\}$ are in 17th qubit $|q_{17}\rangle = |q_{17}\rangle$ named “readout qubit” (see Fig. 3); this step is called quantum encoding [16], [17]. The readout qubit yields either “+1” or “-1” as output. Hence, it is natural to have the two-label Eurosat image data as input image.

Quantum encoding, after the dimensionality reduction of the images, is then the representation of the classic data points \mathbf{v} as quantum states $|q\rangle$. Moreover, we are leveraging the classic

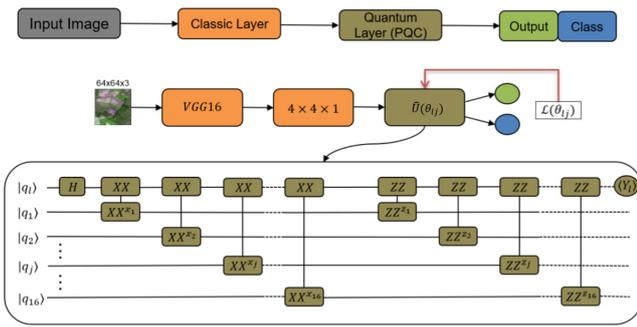


Fig. 3. Top: general scheme for classifying the Eurosat image dataset, Middle: detailed scheme for classifying the Eurosat image dataset, Bottom: architecture of the PQC with inputs of 17 qubits; $|q_l\rangle$ represents a “readout qubit,” and the other ones are “data qubits.” H is a Hadamard matrix, “ XX ” and “ XX^{x_j} ” represent $e^{-i\theta_{ij}^x \hat{X}_l \hat{X}_j}$, and $x_j = \theta_{ij}^x$; the same representation is used for “ ZZ ” and “ ZZ^{z_j} ” [see (5)].

layer to extract the physically meaningful 16 features of the Eurosat images, and then, we are encoding them in 16 qubits of the PQC [see Fig. 3 (top and middle)].

The PQC architecture for the Eurosat image classification task is designed as an energy-based model [2], [18] such that

$$\hat{H} = \sum_j (J_{ij}^x \hat{X}_l \hat{X}_j + J_{ij}^z \hat{Z}_l \hat{Z}_j) \quad (3)$$

where Pauli matrices \hat{X}_l and \hat{Z}_l are representing readout qubit operators, \hat{X}_j and \hat{Z}_j are data qubit operators, and the strength parameters (J_{ij}^x, J_{ij}^z) are among the readout qubit and the data qubit operators.

We then write the unitary operation according to (2) as

$$\hat{U}(\beta) = e^{-i\beta \hat{H}} = e^{-i\beta \sum_j (J_{ij}^x \hat{X}_l \hat{X}_j + J_{ij}^z \hat{Z}_l \hat{Z}_j)}. \quad (4)$$

By noting $\theta_{ij}^x = \beta J_{ij}^x$, $\theta_{ij}^z = \beta J_{ij}^z$, we have

$$\hat{U}(\theta_{ij}) = e^{-i \sum_j (\theta_{ij}^x \hat{X}_l \hat{X}_j + \theta_{ij}^z \hat{Z}_l \hat{Z}_j)} \quad (5)$$

where $(\theta_{ij}^x, \theta_{ij}^z)$ are our training parameters [see Fig. 3 (bottom)]. After the unitary operation expressed by (5) is acted on $|q_j\rangle$, we measure a readout qubit $|q_l\rangle$; the measurement of the readout qubit gives an output, which results from

$$\langle q_l, q_j | \hat{U}^\dagger(\theta_{ij}) \hat{Y}_l \hat{U}(\theta_{ij}) | q_j, q_l \rangle = \langle \hat{Y}_l \rangle, \quad \langle \hat{Y}_l \rangle \in [-1, +1]. \quad (6)$$

Accordingly, the loss function is then defined by

$$\mathcal{L}(\theta_{ij}) = 1 - l(\mathbf{v}) \langle q_l, q_j | \hat{U}^\dagger(\theta_{ij}) \hat{Y}_l \hat{U}(\theta_{ij}) | q_j, q_l \rangle \quad (7)$$

the parameters of which are updated and optimized by using a conventional computer.

In the next section, we introduce a way to reduce the spatial dimension of each image in the Eurosat dataset into $4 \times 4 \times 1$ elements by preserving their feature information and encoding them into 16 qubits.

III. BENCHMARKING THE PQC FOR EARTH OBSERVATION

The Eurosat dataset is a novel dataset composed of ten labels and 27 000 georeferenced images with the RGB spectral

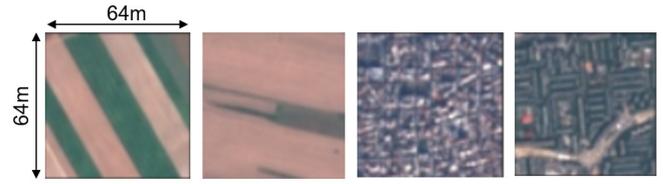


Fig. 4. Eurosat training image with two labels; Annual Crop and Residential.

Algorithm 1 Hybrid Classifier: Classic-Quantum Layer

- 1: **INPUT:** Eurosat data as training data, and Sentinel-2 image of Berlin, Germany as test data.
- 2: **OUTPUT:** Generate two-classes.
- 3: **The dimensionality reduction:** Classic Layer (see Fig. 3):
- 4: Prepare a VGG16 network with the autoencoder.
- 5: Encode the Eurosat data and the image of Berlin to $4 \times 4 \times 1$ elements.
- 6: **Training:** Quantum Layer (see Fig. 3 Bottom):
- 7: Prepare a PQC layer designed as an energy-based model.
- 8: Feed dimensionally-reduced Eurosat data to the PQC layer.
- 9: **Testing:**
- 10: Feed the dimensionally-reduced image of Berlin to the PQC layer.
- 11: **STOP ALGORITHM.**

bands of the Sentinel-2 satellite (see Fig. 4) [14]. Each image within this Eurosat dataset has a size of $64 \times 64 \times 3$ elements and corresponds to a single class. For the classification of this dataset in the PQC, we exploit the hybrid network, namely the classic and the PQC layer (see Fig. 3 and Algorithm 1). The input image to this hybrid network must be the two-label representation of the Eurosat image dataset following the output of the readout qubit in (6).

The classic layer encodes the two-label images with a size of $64 \times 64 \times 3$ elements to ones with a size of $4 \times 4 \times 1$ elements, and then, it maps these encoded images into 16 input qubits by thresholding them with the value of $\tau = 0.5$; this classic layer is hugely motivated by the RBM autoencoder. The PQC layer is a training layer for the two-label image by means of tuning the parameters of the parameterized quantum gates.

A. Our Experiment for Benchmarking the PQC

We designed and implemented our PQC on a classic simulator by using the Tensorflow Quantum python kit. One of the important steps to program this PQC is to choose machinery to feed and embed the classic data into qubits. We had the Eurosat images as our classic data and the classic layer as an autoencoder to feed them to the PQC. Hence, we had to evaluate the performance of the autoencoder for the dimensionality reduction of the Eurosat images. To evaluate its performance, we considered the downsampling technique as well.

For both dimensionality reduction techniques, we classified the dimensionally reduced images on both the PQC and the fully connected classic layer (FCCL) (see Fig. 5).

TABLE I

CLASSIFICATION ACCURACY OF THE PQC AND FCCL FOR THE TWO-LABEL REPRESENTATION OF THE EUROSAT IMAGE DATASET. THE DIMENSION OF THE IMAGES IS REDUCED BY AUTOENCODING OR DOWNSAMPLING; $\{i, j\}$ REPRESENT THE TWO CLASSES; $\{1, 2\} \rightarrow$ ANNUAL CROP AND RESIDENTIAL, $\{3, 4\} \rightarrow$ FOREST AND HIGHWAY, $\{5, 6\} \rightarrow$ FOREST AND INDUSTRIAL, AND $\{7, 8\} \rightarrow$ HERBACEOUS VEGETATION AND RESIDENTIAL

Classes	Binary Classifier Accuracy			
	Autoencoding		Downsampling	
	PQC	FCCL	PQC	FCCL
$\{1, 2\}$	0.9970	0.9994	0.5575	0.4990
$\{3, 4\}$	0.9911	0.9980	0.6230	0.5010
$\{5, 6\}$	0.9995	0.9994	0.7123	0.5010
$\{7, 8\}$	0.9494	0.9789	0.5258	0.5010

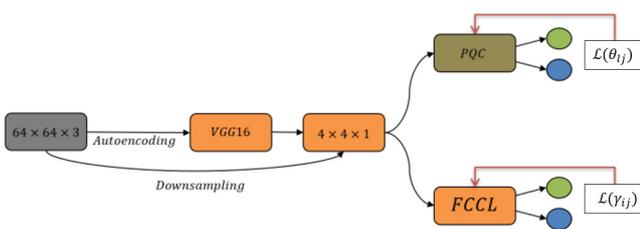


Fig. 5. Graphical representation of the PQC and the classic fully connected classical layer (FCCL) classification for the Eurosat dataset. The dimension of the images is reduced in two ways, by autoencoding and by downsampling.

We performed this experiment for combinations of binary datasets of the Eurosat images, Annual Crop and Residential, Forest and Highway, Forest and Industrial, and Herbaceous Vegetation and Residential.

For the downsampling technique, our experimental results show that the PQC performs better than the FCCL, and it also confirms that the autoencoder is well suited for reducing the dimensionality of the Eurosat image dataset when compared to the downsampling technique (see Table I). Hence, we chose the VGG16 network with the autoencoder for the dimensionality reduction, and we trained the selected hybrid classic-quantum layer for the two-label representation of Eurosat images. Moreover, though the number of qubits is rather limited for the PQC, a classification accuracy of the PQC is even higher than the FCCL in some instances.

These results lead to the conclusion that the PQC could perform even better than conventional classification methods when the number of qubits is increased. In addition, the PQC performance depends on the choice of its optimizer and the depth of the circuit layer, and some care should be taken since the PQC stops a training process due to a vanishing gradient [19]. We experimented on two different optimizers with the Eurosat dataset, namely an Adam optimizer and a root-mean-squared optimizer. With the root-mean-squared optimizer, the PQC stops training the dataset, while the Adam optimizer does not terminate.

IV. VALIDATING THE PQC WITH A SENTINEL-2 IMAGE OF BERLIN, GERMANY

We validated the FCCL and the PQC including its autoencoding with a real-world Sentinel-2 image (see Fig. 5); in particular, we considered a Sentinel-2 image of Berlin,

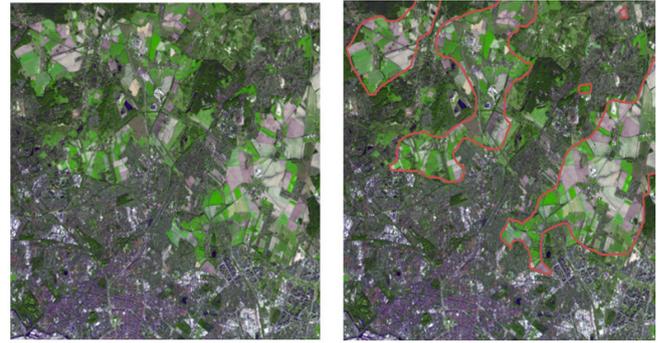


Fig. 6. (Left): Sentinel-2 image of Berlin, Germany, with its RGB spectral band as test image; this test image is not a part of the Eurosat dataset. (Right): we visualized a part of Berlin by the red contour. The inside of this red contour represents the general class Vegetation and Annual Crop, whereas the outside represents all other classes. We classified the inside of this red contour by using the FCCL and the PQC.



Fig. 7. FCCL and PQC results for the classification of the test image of Berlin. The visual images show that the PQC is better for detecting vegetation even within built-up areas than a classic FCCL approach. Top: Annual Crop and Residential. White denotes the Residential class, while black denotes the Annual Crop class. Bottom: Herbaceous Vegetation and Residential. White denotes the Residential class, while black denotes the Herbaceous Vegetation class.

Germany, which is not part of the Eurosat images. This Sentinel-2 image consists of multiple classes instead of two classes [see Fig. 6 (left)]. Thus, we classified several scenes of Berlin for different pairs of class combinations (i.e., two-label examples). When we attempted to classify the two-label examples “Forest and Highway” and “Forest and Industrial,” the two classes were identified as a single class, namely “Forest,” both by the FCCL and the PQC; clearly, the selected Sentinel-2 image of Berlin does not contain the classes “Highway” and “Industrial.” Instead, for the two classes “Annual Crop and

Residential” and “Herbaceous Vegetation and Residential” [see Fig. 6 (right)], the two-label example is classified as two classes as expected. We visualize these results in Fig. 7. Moreover, these visual results demonstrate that the PQC performs and classifies better for a complex remotely sensed image than the FCCL.

V. CONCLUSION

We explored the possible use and challenges encountered with a PQC when handling EO data. More importantly, we assessed the classification performance of a PQC for our EO dataset. First and foremost, we propose a hybrid classic-quantum layer and an autoencoder for dimensionality reduction of Eurosat images. The dimensionality reduction process is even a vital part of classic deep learning methods to reduce the amount of computational effort and to speed up the training process. Second, we encoded the dataset into qubits by thresholding. However, the optimizer has to be chosen wisely to avoid vanishing gradients; we employed an Adam optimizer for training the parameters of the PQC. In addition, we attempted to perform a classification of the two-label NWPU-RESISC45 dataset [20]. Unfortunately, the spatial size of these images is too big for the input of the PQC; in particular, a reduced size with 16 elements by exploiting the autoencoder does not result in meaningful features, in contrast to the Eurosat images. As a first step toward assessing and programming the PQC in EO, such big or high-dimensional images are out of scope currently.

We must note, however, that the proposed method should not be viewed as replacement for classical deep learning methods, but our message is that the classical deep learning methods coexist in unity with PQCs (as advocated in our hybrid quantum-classic network). More importantly, a PQC offers invaluable intuition for constructing a physics-aware (based) deep learning method.

Finally, in terms of future work, we will study the dimensionality reduction techniques for high-dimensional images such as NWPU-RESISC45 for a restricted number of inputs of the PQC, and we will then tackle multilabel classification on the PQC.

ACKNOWLEDGMENT

The authors would like to thank Gottfried Schwartz (DLR Oberpfaffenhofen) for his valuable comments and contributions to enhance their letter.

REFERENCES

- [1] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, pp. 436–444, May 2015, doi: [10.1038/nature14539](https://doi.org/10.1038/nature14539).
- [2] M. Broughton *et al.*, “Tensorflow quantum: A software framework for quantum machine learning,” 2020, *arXiv:2003.02989*. [Online]. Available: <https://arxiv.org/abs/2003.02989>
- [3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017, doi: [10.1145/3065386](https://doi.org/10.1145/3065386).
- [4] R. Salakhutdinov and G. Hinton, “Deep Boltzmann machines,” in *Proc. Int. Conf. Artif. Intell. Statist.*, Clearwater Beach, FL, USA, vol. 5, Apr. 2009, pp. 448–455. [Online]. Available: <http://proceedings.mlr.press/v5/salakhutdinov09a.html>
- [5] M. Kuchhold, M. Simon, and T. Sikora, “Restricted Boltzmann machine image compression,” in *Proc. Picture Coding Symp. (PCS)*, Jun. 2018, pp. 243–247.
- [6] J. Preskill, “Quantum computing in the NISQ era and beyond,” *Quantum*, vol. 2, p. 79, Aug. 2018, doi: [10.22331/q-2018-08-06-79](https://doi.org/10.22331/q-2018-08-06-79).
- [7] V. Dunjko and H. J. Briegel, “Machine learning & artificial intelligence in the quantum domain: A review of recent progress,” *Rep. Prog. Phys.*, vol. 81, no. 7, Jun. 2018, Art. no. 074001, doi: [10.1088/1361-6633/aab406](https://doi.org/10.1088/1361-6633/aab406).
- [8] S. Mangini, F. Tacchino, D. Gerace, D. Bajoni, and C. Macchiavello, “Quantum computing models for artificial neural networks,” *EPL Europhys. Lett.*, vol. 134, no. 1, p. 10002, Apr. 2021, doi: [10.1209/0295-5075/134/10002](https://doi.org/10.1209/0295-5075/134/10002).
- [9] G. Cheng, X. Xie, J. Han, L. Guo, and G.-S. Xia, “Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, no. 99, pp. 3735–3756, Jun. 2020.
- [10] J. Kang, R. Fernandez-Beltran, Z. Ye, X. Tong, P. Ghamisi, and A. Plaza, “Deep metric learning based on scalable neighborhood components for remote sensing scene characterization,” *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8905–8918, Dec. 2020.
- [11] Y. Dong, C. Yang, and Y. Zhang, “Deep metric learning with online hard mining for hyperspectral classification,” *Remote Sens.*, vol. 13, no. 7, p. 1368, Apr. 2021. [Online]. Available: <https://www.mdpi.com/2072-4292/13/7/1368>
- [12] Y. Dong, T. Liang, Y. Zhang, and B. Du, “Spectral–spatial weighted kernel manifold embedded distribution alignment for remote sensing image classification,” *IEEE Trans. Cybern.*, vol. 51, no. 6, pp. 3185–3197, Jun. 2021.
- [13] N. Kussul, M. Lavreniuk, S. Skakun, and A. Shelestov, “Deep learning classification of land cover and crop types using remote sensing data,” *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 5, pp. 778–782, May 2017.
- [14] P. Helber, B. Bischke, A. Dengel, and D. Borth, “EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification,” *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 7, pp. 2217–2226, Jul. 2019.
- [15] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” in *Proc. 3rd Int. Conf. Learn. Represent.*, Y. Bengio and Y. LeCun, Eds., San Diego, CA, USA, May 2015, pp. 1–14. [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [16] S. Lloyd, M. Schuld, A. Ijaz, J. Izaac, and N. Killoran, “Quantum embeddings for machine learning,” 2020, *arXiv:2001.03622*. [Online]. Available: <http://arxiv.org/abs/2001.03622>
- [17] M. Schuld and N. Killoran, “Quantum machine learning in feature Hilbert spaces,” *Phys. Rev. Lett.*, vol. 122, no. 4, Feb. 2019, Art. no. 040504, doi: [10.1103/PhysRevLett.122.040504](https://doi.org/10.1103/PhysRevLett.122.040504).
- [18] E. Farhi and H. Neven, “Classification with quantum neural networks on near term processors,” 2018, *arXiv:1802.06002*. [Online]. Available: <https://arxiv.org/abs/1802.06002>
- [19] A. Skolik, J. R. McClean, M. Mohseni, P. van der Smagt, and M. Leib, “Layerwise learning for quantum neural networks,” *Quantum Mach. Intell.*, vol. 3, no. 1, Jan. 2021, doi: [10.1007/s42484-020-00036-4](https://doi.org/10.1007/s42484-020-00036-4).
- [20] G. Cheng, J. Han, and X. Lu, “Remote sensing image scene classification: Benchmark and state of the art,” *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017.

4.1.2 Publication B: Quantum Transfer Learning for Real-World, Small, and High-Dimensional Remotely Sensed Datasets

S. Otgonbaatar, G. Schwarz, M. Datcu, and D. Kranzlmüller

Abstract. QML models manipulate the initialized qubits of a quantum computer by using parameterized quantum circuits such that the initialized qubits, encoding any input quantum dataset, yield meaningful physical information after measuring them. In particular, QML models only recognize quantum data, while real-world problems involve inherently classical datasets. Hence, **Publication A** proposed and used a hybrid classical-quantum approach and a two-level encoding strategy for embedding ubiquitous classical datasets in the limited number of input qubits. We demonstrated its potential quantum advantage using the large-scale and low-dimensionality Eurosat dataset. However, hybrid classical-quantum approaches still face two challenges regardless of their promised quantum advantage. The two challenges are related to small-scale, high-dimensional real-world datasets and a few input qubits: 1. an embedding challenge due to the high-dimensional datasets and very few input qubits, and 2. a small-scale benchmark dataset challenge compared with a large-scale benchmark dataset. Hence, to tackle these two challenges for benchmarking and validating QML models on small-scale high-dimensional datasets in one go, this work employ quantum transfer learning comprising a classical deep neural network and a multi-qubit parameterized quantum circuit layer, the quantum version of classical transfer learning. Furthermore, we use real-amplitude and strongly-entangling L -layer parameterized quantum circuits with and without data re-uploading layers as a multi-qubit parameterized quantum circuit layer and evaluate their expressive power, quantified by using their local effective dimension; the lower the local effective dimension of a parameterized quantum circuit, the better its performance on unseen data. Our numerical results show that the strongly-entangling L -layer parameterized quantum circuit layer has a lower local effective dimension than the real-amplitude parameterized quantum circuit layer and outperforms it on the hard-to-classify datasets. In addition, quantum transfer learning helps tackle the above-mentioned challenges for benchmarking and validating QML models on small-scale, high-dimensional datasets.

Quantum Transfer Learning for Real-World, Small, and High-Dimensional Remotely Sensed Datasets

Soronzonbold Otgonbaatar, Gottfried Schwarz, Mihai Datcu, *Fellow, IEEE*, and Dieter Kranzlmüller

Abstract—Quantum Machine Learning (QML) models promise to have some computational (or quantum) advantage for classifying supervised datasets (e.g., satellite images) over some conventional Deep Learning (DL) techniques due to their expressive power via their local effective dimension. There are, however, two main challenges regardless of the promised quantum advantage: 1) Currently available quantum bits (qubits) are very small in number, while real-world datasets are characterized by hundreds of high-dimensional elements (*i.e.* features). Additionally, there is not a single unified approach for embedding real-world high-dimensional datasets in a limited number of qubits. 2) Some real-world datasets are too small for training intricate QML networks. Hence, to tackle these two challenges for benchmarking and validating QML networks on real-world, small, and high-dimensional datasets in one-go, we employ quantum transfer learning comprising a classical VGG16 layer and a multi-qubit QML layer. We use real-amplitude and strongly-entangling N-layer QML networks with and without data re-uploading layers as a multi-qubit QML layer, and evaluate their expressive power quantified by using their local effective dimension; the lower the local effective dimension of a QML network, the better its performance on unseen data. As datasets, we utilize Eurosat and synthetic datasets (*i.e.* easy-to-classify datasets), and an UC Merced Land Use dataset (*i.e.* a hard-to-classify dataset). Our numerical results show that the strongly-entangling N-layer QML network has a lower local effective dimension than the real-amplitude QML network and outperforms it on the hard-to-classify datasets. In addition, quantum transfer learning helps tackle the two challenges mentioned above for benchmarking and validating QML networks on real-world, small, and high-dimensional datasets.

Index Terms—quantum transfer learning, quantum machine learning, data re-uploading, Earth observation, remote sensing, image classification.

I. INTRODUCTION

UNIVERSAL quantum computers are composed of a collection of quantum bits (qubits) and parameterized quantum gates being arranged according to some given topologies, while quantum learning algorithms are algorithms manipulating qubits by using parameterized quantum gates. Based on the learnable parameters of parameterized quantum gates, Quantum Machine Learning (QML) as described by [1] and [2], in general, contains three different sub-directions [3]:

- **Quantum-Inspired ML:** develop novel artificial intelligence (AI/DL) techniques by using concepts from quantum information processing [4] and [5].

Soronzonbold Otgonbaatar is with Remote Sensing Technology Institute, German Aerospace Center DLR, Weßling, 82234, Germany and Ludwig-Maximilians-Universität München, Gottfried Schwarz and Mihai Datcu are with the Remote Sensing Technology Institute, German Aerospace Center DLR, Weßling, 82234, Germany, and Dieter Kranzlmüller is with the Ludwig-Maximilians-Universität München.

Corresponding author: S. Otgonbaatar (Soronzonbold.Otgonbaatar@dlr.de)

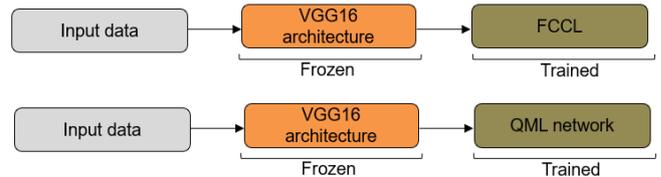


Fig. 1. [Top] classical transfer learning: Input data, a DL VGG16 architecture, a Fully-Connected Classical Layer (FCCL), [Bottom] quantum transfer learning: Input data, a DL VGG16 architecture, a QML network.

- **Quantum ML (sometimes):** apply classical DL techniques to quantum data (quantum chemistry) [6].
- **Quantum-Applied ML:** develop quantum learning circuits for supervised real-world data on quantum computers [7], [8], and [9].

In this work, we focus on Quantum-Applied ML, and we use QML interchangeably with Quantum-Applied ML. A QML network encodes input data in qubits, and learns the parameters of parameterized quantum gates. Moreover, it promises quantum advantage for some computational problems over conventional learning methods due to either its expressive power measured by the local effective dimension [10] and [11], or its computational time [12].

However, the qubits of currently available universal quantum computers are noisy and small in number. Hence, these types of universal quantum computers are called Noisy Intermediate-Scale Quantum (NISQ) computers [13]. Due to the limited number of qubits, there are two main challenges:

- 1) **Embedding challenge:** there is not a single unified approach for embedding real-world, high-dimensional data points in a small number of qubits.
- 2) **Small dataset challenge:** QML networks do not capture informative patterns in small datasets in contrast to big datasets, and this challenge even exists with conventional DL methods.

Here, we name a dataset as a low-dimensional dataset if and only if we can represent its data points by at most five elements using a classical dimensionality-reduction technique, and as a high-dimensional dataset otherwise.

To overcome the *embedding* challenge, some studies already proposed an embedding strategy for a toy dataset [14], as well as for real-world, big, and low-dimensional datasets [9]. In the work of [9], its authors investigated a binary-label *classical-quantum* classifier for embedding and classifying a two-label, low-dimensional Eurosat dataset [15], in which they classified this specific binary dataset by measuring directly the output of

the quantum layer. On the other hand, the article [16] focused on a multi-label *classic-quantum-classic* classifier in which the authors classified the same Eurosat dataset by measuring the last *classical layer* but not a *quantum layer* for a multi-class case. Furthermore, the authors of [17] already also introduced a single- and multi-qubit quantum classifier for embedding a selected practical dataset in a parameterized quantum circuit.

To overcome the *small dataset* challenge, the authors of [18] proposed a novel method named *quantum transfer learning*, the quantum version of classical transfer learning. Classical transfer learning is a procedure for training fully-connected classical layers residing on the top level of conventional DL architectures (for small datasets, when the parameters of frozen conventional DL architectures are initialized a priori on big similar datasets). In contrast, quantum transfer learning trains a QML network placed on the top level of frozen conventional DL architectures instead of fully-connected classical layers (see Fig. 1). In particular, for quantum transfer learning, one replaces the fully-connected top layers of frozen conventional DL architectures by a multi-qubit QML network. One also profits from the advantage of quantum transfer learning since it simultaneously helps tackle the two challenges mentioned above.

For practically important datasets, the authors of [9] and [16] *implicitly* used a quantum transfer learning method, but they employed a real-world, *big*, and *low-dimensional* dataset (*i.e.* an Eurosat dataset). In this work, we *explicitly* propose and employ a quantum transfer learning method for real-world, *small*, and *high-dimensional* datasets. Our proposed quantum transfer learning method consists of a multi-qubit QML network (with or without *data re-uploading*), and a very deep convolutional network (in our case, a VGG16 architecture), playing the role of a feature extractor from datasets as shown in Fig. 1 [Bottom] [19], [20], and [21]. In particular, we employ real-amplitude and strongly-entangling N-layer QML networks with and without *data re-uploading* layers [22] as a multi-qubit QML network, and quantify their expressive power by using their *local effective dimension* (the lower, the better) which gives us a portion of the active parameters in the trained QML network [10], [11], [23], [24] and [25] and currently available quantum resources. The expressive power is referred to as the capacity of any learning model, and its capability to capture intricate patterns in any dataset. Moreover, the lower the local effective dimension of a given QML network, the better its performance on still unseen data points.

Our practical datasets are synthetic, Eurosat, and UC Merced Land Use images [26]. First, we compute the local effective dimension of our QML networks, that is, real-amplitude and strongly-entangling N-layer QML networks without *data re-uploading* layers, on two-class low-dimensional synthetic and Eurosat datasets since we can compress and represent these low-dimensional datasets by 3 and 4 elements using a classical DL network as proposed by [9], respectively; here, we generated and used two-class synthetic data including 100 data points, where each data point is a two-dimensional vector drawn from circular data with the error according to a normal distribution $\mathcal{N}(\mu = 0, \sigma = 1)$. For low-dimensional Eurosat data, we utilized its *annual crop*

and *residential* area classes, and this two-class set consists of 4,000 data points, each of which is characterized by $64 \times 64 \times 3$ low-dimensional elements. To validate the relationship between local effective dimension and performance (*i.e.* the classification accuracy) of a given QML network, we trained, subsequently, our QML networks via quantum transfer learning on the small, high-dimensional, and hard-to-classify three-class images, *i.e.* *dense residential*, *medium residential*, and *sparse residential* area classes of the high-dimensional UC Merced Land Use dataset. Additionally, a *dense residential*, *medium residential*, and *sparse residential* area classification problem meets our two challenges mentioned, because this three-class images consist of only 288 data points, and each image is characterized by $256 \times 256 \times 3$ high-dimensional elements [26] and [27]. As a quantum simulator, we used a PennyLane Python library for training our QML networks [28].

Our experimental results demonstrate that the strongly-entangling N-layer QML network without *data re-uploading* layers has a lower local effective dimension than the real-amplitude N-layer QML network without *data re-uploading* layers. It also has a higher test accuracy on real-world datasets than the real-amplitude QML network. Furthermore, quantum transfer learning helps tackle the two above-mentioned challenges for benchmarking and validating different QML networks on real-world, small, high-dimensional, and hard-to-classify datasets.

This work is structured as follows: in Sections II and III, we provide some background for quantum transfer learning and multi-qubit QML networks, respectively. In Section IV, we present the expressive power of QML networks via their local effective dimension. Subsequently, in Section V, we introduce practical datasets being used in this paper. In Section VI, we present our experiments and some of our findings. Finally, we draw a few conclusions in Section VII.

II. QUANTUM TRANSFER LEARNING

Quantum transfer learning is referred to as training a QML network with and without *data re-uploading* on real-world small datasets when the weights of the VGG16 architecture are initialized on the *ImageNet* dataset. Moreover, we froze the weights of the VGG16 such that none of its weights were updated during the training of a QML network. The QML network we propose in this study is a multi-qubit N-layer quantum classifier (with and without *data re-uploading*); these classifiers are extremely simple as well as very powerful learning networks for non-linear datasets [19] and [20].

A. Single-Qubit QML Network with and without Data Re-Uploading

We characterize a single-qubit QML network by a tensorial feature map and universal parameterized quantum gates:

$$U(\phi_1, \phi_2, \phi_3)V_i(\theta_{i,1}, \gamma_{i,2}, \phi_{i,3}). \quad (1)$$

where ϕ_1, ϕ_2 , and ϕ_3 embed input data points in qubits, and $\theta_{i,1}, \gamma_{i,2}$, and $\phi_{i,3}$ are learning weights at the i th layer.

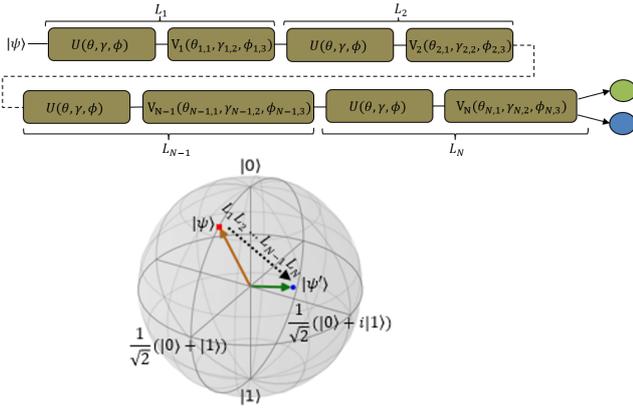


Fig. 2. [Top] A single-qubit N-layer QML network with *data re-uploading*, [Bottom] A QML network operation on a single-qubit represented on a Bloch sphere, in which $|\psi\rangle$ is an input single-qubit, and $|\psi'\rangle$ is the single-qubit after $L_1, L_2, \dots, L_{N-1}, L_N$ quantum layers.

Namely, we encode an input data point having three features in qubits using a tensorial feature map, $U(\phi_1, \phi_2, \phi_3) = U(\phi_1) \otimes U(\phi_2) \otimes U(\phi_3)$, and subsequently, we train a parameterized quantum gate $V_i(\theta_{i,1}, \gamma_{i,2}, \phi_{i,3})$ on the embedded data points. For simplicity, we express $U = U(\phi_1, \phi_2, \phi_3)$ and $V_i = V_i(\theta_{i,1}, \gamma_{i,2}, \phi_{i,3}) = V_i(\theta_{i,1})V_i(\gamma_{i,2})V_i(\phi_{i,3})$ since they are unitary quantum gates such that $V_i = V_i^\dagger, i = 1, \dots, N$ where N is the number of layers or the depth of a given QML network [17]. In matrix form, we express U and the V_i 's by:

$$U(\phi_j) = e^{i\phi_j} \begin{pmatrix} \cos(\phi_j) & -\sin(\phi_j) \\ \sin(\phi_j) & \cos(\phi_j) \end{pmatrix}, \quad j = 1, 2, 3, \quad (2)$$

and

$$V_i(\theta_{i,1}) = \begin{pmatrix} e^{i\theta_{i,1}} & 0 \\ 0 & e^{-i\theta_{i,1}} \end{pmatrix}, \quad V_i(\gamma_{i,2}) = \begin{pmatrix} e^{i\gamma_{i,2}} & 0 \\ 0 & e^{-i\gamma_{i,2}} \end{pmatrix}, \quad (3)$$

and

$$V_i(\phi_{i,3}) = \begin{pmatrix} \cos(\phi_{i,3}) & -\sin(\phi_{i,3}) \\ \sin(\phi_{i,3}) & \cos(\phi_{i,3}) \end{pmatrix}, \quad i = 1, \dots, N. \quad (4)$$

For a QML network with *data re-uploading*, the quantum circuit UV_i expressed by Eq. (1) is repeated N times:

$$\prod_{i=1}^N UV_i = \underbrace{UV_1}_{L_1} \underbrace{UV_2}_{L_2} \dots \underbrace{UV_{N-1}}_{L_{N-1}} \underbrace{UV_N}_{L_N}, \quad (5)$$

where an input data point is re-uploaded N times in the quantum gate U , and the parameterized quantum gate V_N is also repeated N times as shown in Fig. 2. The N^{th} layer is denoted as L_N .

Alternatively, for a QML network without *data re-uploading*, the quantum circuit UV_i expressed in Eq. (1) is repeated N times:

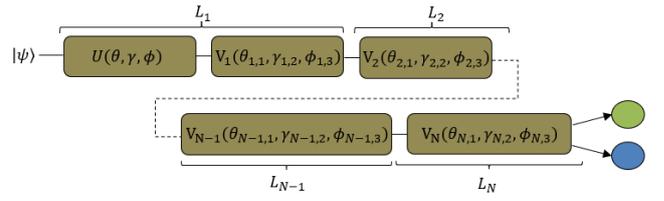


Fig. 3. A single-qubit N-layer QML network without *data re-uploading*

$$U \prod_{i=1}^N V_i = \underbrace{UV_1}_{L_1} \underbrace{V_2}_{L_2} \dots \underbrace{V_{N-1}}_{L_{N-1}} \underbrace{V_N}_{L_N}, \quad (6)$$

where an input data point is uploaded in U only once at layer L_1 as shown in Fig. 3.

For single-qubit QML networks, a qubit is represented by:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}, \quad (7)$$

and as a bra vector $\langle 0| = |0\rangle^\dagger, \langle 1| = |1\rangle^\dagger$, where \dagger represents both transpose and conjugate. In general, qubits can exist in a superposition:

$$|\psi\rangle = c_1 |0\rangle + c_2 |1\rangle \quad \text{such that} \quad |c_1|^2 + |c_2|^2 = 1, \quad (8)$$

where c_1 and c_2 are complex numbers as shown in Fig. 2 [Bottom].

Let us consider a simple example for a single-qubit *data re-uploading* QML network: we assume $|\psi\rangle = |1\rangle$ and a QML network with one layer L_1 , $(0, 0, \phi_3) = (0, 0, \pi/2)$ and $(\theta_{1,1}, \gamma_{1,2}, \phi_{1,3}) = (0, 0, 0)$. A qubit after layer L_1 becomes:

$$|\psi'\rangle = V_1(0, 0, 0)U(\pi/2)|1\rangle. \quad (9)$$

If we measure a single-qubit by using a projective measurement $\hat{z} = |0\rangle\langle 0| - |1\rangle\langle 1|$, (*i.e.* the outer product of $|0\rangle$ and $|1\rangle$), we obtain an expected value of measuring the state $|\psi'\rangle$ in the basis \hat{z} :

$$\langle \hat{z} \rangle_{\vec{\theta}} = \langle \psi' | \hat{z} | \psi' \rangle = e^{-i\pi} e^{i\pi} \langle 0 | \hat{z} | 0 \rangle = 1. \quad (10)$$

where $\vec{\theta} = (\theta_{i,1}, \gamma_{i,2}, \phi_{i,3})^T \in \Theta$ and T denotes the transpose operation of a vector. In the end, this expected value is connected to two classical neurons shown in Fig. 2, and these neurons denoted as $l = 1, 2$ are defined by:

$$\tilde{y}_l = A(a_l + w_l \langle \hat{z} \rangle_{\vec{\theta}}), \quad (11)$$

where $A(\cdot)$ is a non-linear activation function, that is, a sigmoid function, \tilde{y}_l is their predicted output, a_l is their bias, and w_l is their edge parameter [17].

In general, we express the expected value of a single-qubit N-layer QML network with and without *data re-uploading* into the basis $\hat{z} = |0\rangle\langle 0| - |1\rangle\langle 1|$ by:

$$\langle \hat{z} \rangle_{\vec{\theta}} = \langle \psi | L_1^\dagger L_2^\dagger \dots L_{N-1}^\dagger L_N^\dagger \hat{z} L_N L_{N-1} \dots L_2 L_1 | \psi \rangle \quad (12)$$

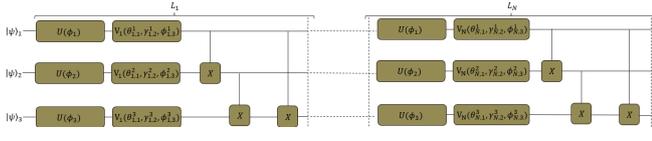


Fig. 4. A multi-qubit N-layer QML network with *data re-uploading*: L_1, \dots, L_N quantum layers.

yielding a continuous value from -1 to $+1$. We then connect two classical neurons to the expected value $\langle \hat{z} \rangle_{\vec{\theta}}$ as shown in Eq. (11).

III. OUR MULTI-QUBIT QML NETWORKS WITH AND WITHOUT DATA RE-UPLOADING

A multi-qubit QML network is a parameterized quantum circuit (PQC) with some input qubits and parameterized quantum gates exploiting Eq. (2) and Eq. (3). We encoded our data in its input qubits by utilizing a tensorial feature map expressed by Eq. (2) as proposed by [14], [19], and [20], and we trained its parameterized quantum gates following Eq. (3). Moreover, we name a multi-qubit QML network with N layers as a multi-qubit N-layer QML network similar to the single-qubit N-layer QML network mentioned in the previous section.

In this work, we use three-qubit real-amplitude and strongly-entangling N-layer QML networks with and without *data re-uploading* layers. We express the parameterized quantum gates of these QML networks with *data re-uploading* layers by:

$$\prod_{i=1}^N \underbrace{U(\phi_1, \phi_2, \phi_3) V_i(\theta_{i,1}^q, \gamma_{i,2}^q, \phi_{i,3}^q) W_i}_{L_i}, \quad q = 1, 2, 3, \quad (13)$$

where q represents the qubit number $|\psi\rangle_q$, and each W_i denotes controlled-X quantum gates (see Fig. 4). A controlled-X quantum gate is a two-qubit quantum gate acting on a target qubit if and only if a control qubit is in the state $|1\rangle$. In contrast, these QML networks without *data re-uploading* layers are characterized by:

$$U(\phi_1, \phi_2, \phi_3) \prod_{i=1}^N \underbrace{V_i(\theta_{i,1}^q, \gamma_{i,2}^q, \phi_{i,3}^q) W_i}_{L_i}, \quad (14)$$

where the classical data is uploaded once at the layer L_1 as illustrated in Fig. 5.

A. Real-Amplitude N-Layer QML Networks with and without Data Re-Uploading Layers

Our real-amplitude N-layer QML networks with and without *data re-uploading* layers are composed of a quantum gate U encoding a data point in qubits, parameterized quantum gates V_i , and non-parameterized quantum gates W_i , respectively:

$$\prod_{i=1}^N \underbrace{U(\phi_1, \phi_2, \phi_3) V_i(0, 0, \phi_{i,3}^q) W_i}_{L_i} \quad (15)$$

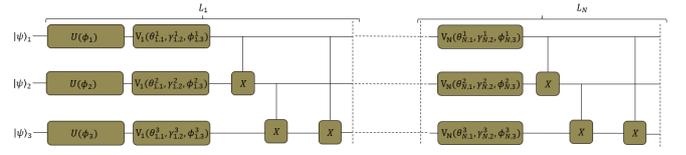


Fig. 5. A multi-qubit N-layer QML network without *data re-uploading*: L_1, \dots, L_N quantum layers.

and

$$U(\phi_1, \phi_2, \phi_3) \prod_{i=1}^N \underbrace{V_i(0, 0, \phi_{i,3}^q) W_i}_{L_i}, \quad (16)$$

where W_i represents all-to-all entanglement [16].

B. Strongly-Entangling N-Layer QML Networks with and without Data Re-Uploading Layers

Our strongly-entangling N-layer QML networks with and without *data re-uploading* layers are defined by:

$$\prod_{i=1}^N \underbrace{U(\phi_1, \phi_2, \phi_3) V_i(\theta_{i,1}^q, \gamma_{i,2}^q, \phi_{i,3}^q) W_i}_{L_i} \quad (17)$$

and

$$U(\phi_1, \phi_2, \phi_3) \prod_{i=1}^N \underbrace{V_i(\theta_{i,1}^q, \gamma_{i,2}^q, \phi_{i,3}^q) W_i}_{L_i}, \quad (18)$$

where W_i represents strong entanglement [22].

IV. THE POWER OF QUANTUM MACHINE LEARNING NETWORKS

Deep neural networks are powerful learning models for identifying intricate patterns in big datasets. Their power is quantified by using the so-called local effective dimension which yields a portion of active parameters in the trained network [11] and [24]. Quantum neural networks, *i.e.* QML networks, are novel learning models based on PQCs exploiting quantum superposition, entanglement, and the interference of qubits. The authors of [10] and [11] demonstrated that some QML networks have a lower local effective dimension – a lower local effective dimension and being more powerful – for analyzing some data-driven tasks than their classical counterparts. In particular, the local effective dimension gives us the active parameters $\vec{\theta}^*$ in the trained model, and the lower effective dimension of a learning model (classical or quantum), the better it generalizes on unseen data points.

According to [10], [11], and [29], the local effective dimension of a QML network $\langle \hat{z} \rangle_{\vec{\theta}}$ around the active parameters $\vec{\theta}^* \in \Theta \subset \mathbb{R}^d$ with n training data points, where Θ weight space, is defined by:

$$d_{n,\lambda}(\langle \hat{z} \rangle_{\vec{\theta}}) = \frac{2 \log \left(\frac{1}{V_\epsilon} \int_{\mathcal{B}_\epsilon(\vec{\theta}^*)} \sqrt{\det(I_d + k_{n,\lambda} \bar{F}(\vec{\theta}) d\vec{\theta})} \right)}{\log k_{n,\lambda}}, \quad (19)$$

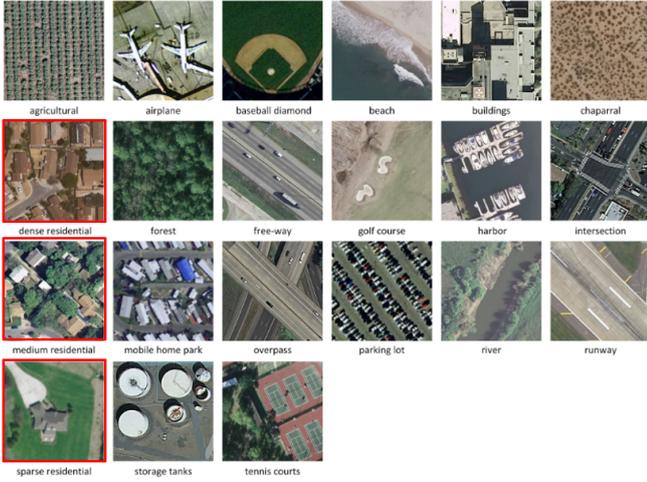


Fig. 6. Some examples of the high-dimensional UC Merced Land Use dataset taken from [26]. Here, we contoured its hard-to-classify three-class examples in red.

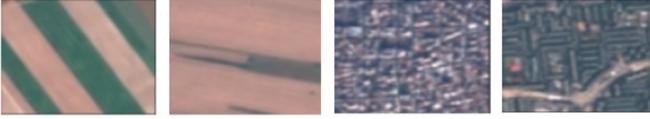


Fig. 7. Annual crop and residential area classes of the low-dimensional Eurosat dataset.

where $\mathcal{B}_\epsilon(\vec{\theta}^*) := \{\vec{\theta} \in \Theta : \|\vec{\theta} - \vec{\theta}^*\| \leq \epsilon\}$ is an ϵ -ball with a volume V_ϵ , $\epsilon > 1/\sqrt{n}$, $k_{n,\lambda} = \frac{\lambda n}{2\pi \log n}$, $\lambda \in \left(\frac{2\pi \log n}{n}, 1\right]$, I_d is a unit diagonal matrix, and $\bar{F}(\vec{\theta}) \in \mathbb{R}^{d \times d}$ is the normalized Fisher information matrix of $F(\vec{\theta})$ [30] with an element:

$$\bar{F}_{ij}(\vec{\theta}) = \frac{d \cdot V_\epsilon}{\int_{\mathcal{B}_\epsilon(\vec{\theta}^*)} \text{Tr}(F(\vec{\theta})) d\vec{\theta}} F_{ij}(\vec{\theta}). \quad (20)$$

V. OUR DATASETS

We generated and used a synthetic dataset using

$$x_m = r_m \begin{pmatrix} \cos \phi_m \\ \sin \phi_m \end{pmatrix} + \begin{pmatrix} \epsilon_m^x \\ \epsilon_m^y \end{pmatrix} \quad (21)$$

where $r_m = 1$ if $y_m = 1$, and $r_m = 0.15$ if $y_m = 0$, ϵ_m 's are a normal distribution $\mathcal{N}(\mu = 0, \sigma = 1)$, and $\phi_n \in (0, 2\pi]$ linearly spaced [31]. This synthetic dataset is a two-class dataset composed of $m = 100$ data points.

A Eurosat dataset is a Sentinel-2 image dataset with 27,000 labelled and georeferenced images. Additionally, this dataset is a patch-based dataset with 64×64 pixels comprising 10 classes, where each image is characterized by 13 spectral bands ranging from 443 nm to 2190 nm, and having a spatial resolution of 60 m/pixel (see Fig. 7). We used selected two-class image scenes, namely the *annual crop* and *residential* area classes, consisting of 4,000 images each of which is characterized by $64 \times 64 \times 3$ low-dimensional elements.

UC Merced Land Use data contain image scenes of 21 classes with three RGB spectral bands. Its smallest class,

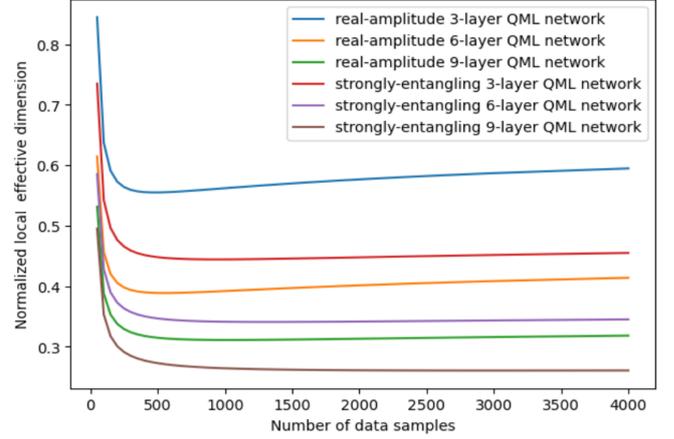


Fig. 8. Normalized local effective dimension of real-amplitude and strongly-entangling N-layer QML networks without *data re-uploading* with respect to the number of data samples.

tennis court, includes 42 data points, while its largest classes (e.g., *agricultural*), include 100 data points. In total, this dataset comprises 2100 image scenes [26], [27]. By visual inspection and experiment, we learned that the hard-to-classify three-class examples of this dataset are the *dense residential*, *medium residential*, and *sparse residential* area classes shown in Fig. 6. For our tests, we employed this hard-to-classify three-class examples comprising 288 image scenes, where each image is a $256 \times 256 \times 3$ high-dimensional elements.

VI. OUR EXPERIMENTS

In this section, we define the local effective dimension and the classification performance of our QML networks, that is, real-amplitude and strongly-entangling N-layer QML networks, while utilizing synthetic and Eurosat datasets. In particular, we derive the relationship between the local effective dimension and the classification accuracy of our QML networks on synthetic and real-world datasets. In order to validate the relationship between the local effective dimension and the classification power of our QML networks, we train them via quantum transfer learning on the real-world, small, and high-dimensional three-class dataset (the hard-to-classify dataset), *i.e.* *dense residential*, *medium residential*, and *sparse residential* area classes, of the UC Merced Land Use dataset, because this three-class dataset meets the two challenges mentioned before:

- 1) Embedding challenge: the three-class images are characterized by $256 \times 256 \times 3$ high-dimensional elements, while currently available quantum computers already have around 50 qubits.
- 2) Small dataset challenge: the three-class dataset consists of only 288 image scenes, and we split it into a training dataset comprising 201 image scenes, and a test dataset consisting of 87 image scenes.

In particular, we benchmarked and validated our QML networks placed on the top level of the frozen VGG16 network on the hard-to-classify three-class dataset, because these datasets play a very important role in QML [32].

TABLE I
CLASSIFICATION ACCURACY OF REAL-AMPLITUDE AND STRONGLY-ENTANGLING N-LAYER QML NETWORKS WITHOUT DATA RE-UPLOADING ON THE TWO-CLASS SYNTHETIC DATASET

{Class}/N-layer	real-amplitude			strongly-entangling		
	3	6	9	3	6	9
{1, 2}	0.35	0.45	0.40	0.90	0.95	1.00

TABLE II
CLASSIFICATION ACCURACY OF N-DEPTH REAL-AMPLITUDE AND STRONGLY-ENTANGLING NETWORKS WITHOUT A DATA RE-UPLOADING LAYER ON THE ANNUAL CROP AND RESIDENTIAL AREA CLASSES OF THE EUROSAT DATASET; HERE, {1, 2} REPRESENTS THE ANNUAL CROP AND RESIDENTIAL AREA CLASSES.

{Class}/N-depth	real-amplitude			strongly-entangling		
	3	6	9	3	6	9
{1, 2}	0.70	0.73	0.75	0.74	0.72	0.70

TABLE III
CLASSIFICATION ACCURACY OF N-DEPTH REAL-AMPLITUDE AND STRONGLY-ENTANGLING NETWORKS WITHOUT A DATA RE-UPLOADING LAYER ON THE DENSE RESIDENTIAL, MEDIUM RESIDENTIAL, AND SPARSE RESIDENTIAL AREA CLASSES OF THE UC MERCED LAND USE DATASET; HERE, {1, 2, 3} REPRESENTS THE DENSE RESIDENTIAL, MEDIUM RESIDENTIAL, AND SPARSE RESIDENTIAL AREA CLASSES.

{Class}/N-depth	real-amplitude			strongly-entangling		
	3	6	9	3	6	9
{1, 2, 3}	0.64	0.67	0.72	0.81	0.83	0.83

TABLE IV
CLASSIFICATION ACCURACY OF N-DEPTH REAL-AMPLITUDE AND STRONGLY-ENTANGLING NETWORKS WITH A DATA RE-UPLOADING LAYER ON THE DENSE RESIDENTIAL, MEDIUM RESIDENTIAL, AND SPARSE RESIDENTIAL AREA CLASSES OF THE UC MERCED LAND USE DATASET; HERE, {1, 2, 3} REPRESENTS THE DENSE RESIDENTIAL, MEDIUM RESIDENTIAL, AND SPARSE RESIDENTIAL AREA CLASSES.

{Class}/N-depth	real-amplitude			strongly-entangling		
	3	6	9	3	6	9
{1, 2, 3}	0.41	0.52	0.48	0.74	0.66	0.74

A. Local Effective Dimension and Performance of our QML Networks without Data Re-Uploading

In our numerical study, we employed N-layer QML networks with only three and four qubits in order to keep the quantum resources as low as possible when $N = 3, 6$, or 9. For the synthetic dataset, we computed the local effective dimension of real-amplitude and strongly-entangling N-layer QML networks without *data re-uploading* while increasing the size of the dataset. We plotted the local effective dimension of our QML networks with respect to the dataset size shown in Fig. 8 and present their classification accuracy in Table I. This result leads to the conclusion that the local effective dimension of QML networks seems to correlate with their classification accuracy. Furthermore, we discovered that strongly-entangling N-layer QML networks have a lower effective dimension and, at the same time, a higher accuracy for generating two-class labels than their counterpart real-amplitude QML networks.

To validate this conclusion for a real-world, big, and low-dimensional dataset comprising the *annual crop* and *residential* area classes of the Eurosat dataset, we first applied a so-called two-level encoding which maps each image scene

Algorithm 1 Quantum transfer learning for the hard-to-classify three-class images on the PennyLane simulator

- 1: INPUT: The hard-to-classify three-class examples
- 2: OUTPUT: The hard-to-classify three-class labels
- 3: QUANTUM TRANSFER LEARNING: a sequential model having the frozen VGG16 layer for extracting informative features from remotely-sensed datasets, and a three-qubit QML layer for training on the output of the frozen VGG16 layer (see Fig. 1 [Bottom])
- 4: TRAINING PARAMETERS: epochs=20, batch=64, and the Adam optimizer having the learning rate of 10^{-4} .
- 5: STOP ALGORITHM

characterized by $64 \times 64 \times 3$ low-dimensional elements to 2×2 informative features using the VGG16 architecture, and encoded these informative features in four qubits by employing a tensorial feature map [9]. Then we calculated numerically the local effective dimension of the real-amplitude and strongly-entangling N-layer QML networks with a 3200-element training set and an 800-element testing set randomly sampled from the *annual crop* and *residential* area classes. We found that the strongly-entangling N-layer QML network generates two-class labels similar (and even better in some instances) to the ones generated by the real-amplitude N-layer QML network (see Table II), because the *annual crop* and *residential* area classes have less overlap which is proven by generating two-class labels and by visual inspection. In our case, this two-class classification problem is an easy-to-classify two-class labelling problem, though the strongly-entangling N-layer QML network has a lower local effective dimension and is more powerful than the real-amplitude N-layer QML network.

B. Quantum Transfer Learning for Real-World, Small, High-Dimensional, and Hard-to-Classify Three-Class Images

We validated the relationship between the local effective dimension and the classification accuracy of our QML networks with and without *data re-uploading* on the real-world, small, and high-dimensional three-class images (*i.e.* hard-to-classify images) of the UC Merced Land Use dataset, that is, *dense residential*, *medium residential*, and *sparse residential* area classes. To validate this relationship illustrated in Fig. 8, we placed our QML networks with three input qubits on the top layer of the VGG16 network, since we applied our hard-to-classify three-class images split into a 201-element training set and an 87-element testing set, where each data point is characterized by $256 \times 256 \times 3$ high-dimensional elements compared with the big, low-dimensional Eurosat dataset. We summarized our results in Tables III and IV. Here, we could prove that strongly-entangling N-layer networks without *data re-uploading* layers outperform real-amplitude N-layer networks without *data re-uploading* layers in most instances, except for the $N = 9$ case due to their lower local effective dimension (see Table III and Fig. 8). The poor performance of the strongly-entangling 9-layer QML network without *data re-uploading* layers is caused by their vanishing gradient

[33], that is beyond the main scope of this article but an important future research direction. Moreover, real-amplitude and strongly-entangling N-layer networks without *data re-uploading* layers outperform (on practical high-dimensional datasets) ones with *data re-uploading* layers on the same dataset (see Tables III and IV).

VII. CONCLUSION

We employed and benchmarked real-amplitude and strongly-entangling N-layer QML networks with and without *data re-uploading* layers. As practical datasets, we used a two-class synthetic dataset, easy-to-classify two-class images of the Eurosat dataset, and hard-to-classify three-class images of the UC Merced Land Use dataset. The hard-to-classify three-class dataset consists of 288 image scenes, where each image scene has $256 \times 256 \times 3$ high-dimensional elements, while the easy-to-classify two-class images are composed of 4,000 image scenes each of which is characterized by $64 \times 64 \times 3$ low-dimensional elements. In particular, our hard-to-classify three-class images meet the above-mentioned two challenges for training multi-qubit QML networks: the embedding and the small dataset challenge.

We analyzed the expressive power of real-amplitude and strongly-entangling N-layer QML networks without *data re-uploading* layers via their so-called local effective dimension, while utilizing the synthetic and Eurosat datasets. Our numerical experiments proved the statement that the lower the local effective dimension of a multi-qubit QML network, the better its classification accuracy on unseen data points. More importantly, we discovered that our strongly-entangling N-layer QML networks have a lower local effective dimension and a higher test accuracy than real-amplitude QML networks (see Fig. 8 and Table I). Here, the local effective dimension of QML networks seems to correlate with their classification performance. We note, however, that for easy-to-classify datasets (in our case, two-class labels of the low-dimensional Eurosat dataset), multi-qubit QML networks perform equally well on unseen data points even though one of them is more powerful than other ones (see Table II). Thus, the hard-to-classify datasets are very important datasets for benchmarking and validating QML networks [32].

To validate the relationship between the local effective dimension and the classification performance of real-amplitude and strongly-entangling N-layer QML networks, we trained them via quantum transfer learning on the real-world, small, and hard-to-classify three-class images of the high-dimensional UC Merced Land Use dataset. Our experimental results demonstrate that strongly-entangling N-layer QML networks perform better than real-amplitude N-layer QML networks in most instances (see Table III). Furthermore, real-amplitude and strongly-entangling N-layer QML networks without *data re-uploading* outperform ones with *data re-uploading*. More importantly, quantum transfer learning even helps tackle the two main challenges in one-go encountered for benchmarking and validating multi-qubit QML networks on real-world, small, and high-dimensional datasets of practical importance.

We must note, however, that our message is that we did not attempt to demonstrate computational advantage of QML models over their conventional counterparts, which is already demonstrated by the authors of [9], [10], [11], and [23] but design and select a powerful model among existing QML models for real-world problems of practical importance. Our contribution is also two-fold: I) we designed QML models and analyzed their expressive power via their local effective dimension, since their local effective dimension correlates with their classification capability, and II) we proposed and utilized quantum transfer learning for benchmarking and analyzing QML models on real-world, hard-to-classify datasets, because weaker models generate similar performance metrics (e.g., classification accuracy or loss) on real-world, easy-to-classify datasets as more powerful models.

For ongoing and future work, we will integrate faster and simpler QML models with artificial intelligence methodologies based on their quantum resource required [34]. Plus, we will invent and design quantum-inspired networks for practical and significant problems to obtain quantum advantage as early and efficiently as possible. More importantly, quantum and quantum-inspired models help boost conventional probabilistic models for remotely-sensed datasets. In addition to the computational advantage of a quantum computer, another advantage of quantum computing to remote sensing is that quantum machine learning algorithms are operate inherently on complex vector space [35], and some remote sensing datasets are complex-numbered images. Therefore, we design inherently complex quantum machine learning algorithms for complex-numbered remote sensed images like synthetic aperture radar (SAR) images without the need of modification in conventional machine and deep learning techniques operating on real number space.

ACKNOWLEDGMENTS

We would like to acknowledge Begüm Demir (Technical University of Berlin) for her valuable comments and suggestions on the very first draft of this paper.

REFERENCES

- [1] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, Sep 2017. [Online]. Available: <https://doi.org/10.1038/nature23474>
- [2] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Physics*, vol. 10, no. 9, pp. 631–633, Sep 2014. [Online]. Available: <https://doi.org/10.1038/nphys3029>
- [3] V. Dunjko and H. J. Briegel, "Machine learning & artificial intelligence in the quantum domain: a review of recent progress," *Reports on Progress in Physics*, vol. 81, no. 7, p. 074001, Jun 2018. [Online]. Available: <https://doi.org/10.1088/1361-6633/aab406>
- [4] E. M. Stoudenmire and D. J. Schwab, "Supervised learning with quantum-inspired tensor networks," 2016. [Online]. Available: <https://arxiv.org/abs/1605.05775>
- [5] T. Felser, M. Trenti, L. Sestini, A. Gianelle, D. Zuliani, D. Lucchesi, and S. Montangero, "Quantum-inspired machine learning on high-energy physics data," *npj Quantum Information*, vol. 7, no. 1, p. 111, Jul 2021. [Online]. Available: <https://doi.org/10.1038/s41534-021-00443-w>
- [6] P. Zheng, R. Zubatyuk, W. Wu, O. Isayev, and P. O. Dral, "Artificial intelligence-enhanced quantum chemical method with broad applicability," *Nature Communications*, vol. 12, no. 1, p. 7022, Dec 2021. [Online]. Available: <https://doi.org/10.1038/s41467-021-27340-2>

- [7] D. Willsch, M. Willsch, H. De Raedt, and K. Michielsen, "Support vector machines on the *D-Wave* quantum annealer," *Computer Physics Communications*, vol. 248, p. 107006, 2020. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S001046551930342X>
- [8] C. Tüysüz, C. Rieger, K. Novotny, B. Demirköz, D. Dobos, K. Potamianos, S. Vallecorsa, J.-R. Vlimant, and R. Forster, "Hybrid quantum classical graph neural networks for particle track reconstruction," *Quantum Machine Intelligence*, vol. 3, no. 2, p. 29, Nov 2021. [Online]. Available: <https://doi.org/10.1007/s42484-021-00055-9>
- [9] S. Otgonbaatar and M. Datcu, "Classification of remote sensing images with parameterized quantum gates," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.
- [10] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, "The power of quantum neural networks," *Nature Computational Science*, vol. 1, no. 6, pp. 403–409, Jun 2021. [Online]. Available: <https://doi.org/10.1038/s43588-021-00084-1>
- [11] A. Abbas, D. Sutter, A. Figalli, and S. Woerner, "Effective dimension of machine learning models," 2021. [Online]. Available: <https://arxiv.org/abs/2112.04807>
- [12] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Phys. Rev. Lett.*, vol. 113, p. 130503, Sep 2014. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevLett.113.130503>
- [13] J. Preskill, "Quantum Computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018. [Online]. Available: <https://doi.org/10.22331/q-2018-08-06-79>
- [14] M. Schuld and N. Killoran, "Quantum machine learning in feature hilbert spaces," *Phys. Rev. Lett.*, vol. 122, p. 040504, Feb 2019. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevLett.122.040504>
- [15] P. Helber, B. Bischke, A. Dengel, and D. Borth, "EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 12, no. 7, pp. 2217–2226, 2019.
- [16] A. Sebastianelli, D. A. Zaidenberg, D. Spiller, B. L. Saux, and S. L. Ullo, "On circuit-based hybrid quantum neural networks for remote sensing imagery classification," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 15, pp. 565–580, 2022.
- [17] S. Otgonbaatar and M. Datcu, "Natural embedding of the Stokes parameters of polarimetric synthetic aperture radar images in a gate-based quantum computer," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–8, 2022.
- [18] A. Mari, T. R. Bromley, J. Izaac, M. Schuld, and N. Killoran, "Transfer learning in hybrid classical-quantum neural networks," *Quantum*, vol. 4, p. 340, Oct. 2020. [Online]. Available: <https://doi.org/10.22331/q-2020-10-09-340>
- [19] A. Pérez-Salinas, A. Cervera-Lierta, E. Gil-Fuster, and J. I. Latorre, "Data re-uploading for a universal quantum classifier," *Quantum*, vol. 4, p. 226, Feb. 2020. [Online]. Available: <https://doi.org/10.22331/q-2020-02-06-226>
- [20] S. Lloyd, M. Schuld, A. Ijaz, J. Izaac, and N. Killoran, "Quantum embeddings for machine learning," May 2020. [Online]. Available: <https://arxiv.org/abs/2001.03622>
- [21] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [22] M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, "Circuit-centric quantum classifiers," *Phys. Rev. A*, vol. 101, p. 032308, Mar 2020. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevA.101.032308>
- [23] T. Haug, K. Bharti, and M. Kim, "Capacity and quantum geometry of parametrized quantum circuits," *PRX Quantum*, vol. 2, p. 040309, Oct 2021. [Online]. Available: <https://link.aps.org/doi/10.1103/PRXQuantum.2.040309>
- [24] W. J. Maddox, G. Benton, and A. G. Wilson, "Rethinking parameter counting in deep models: Effective dimensionality revisited," 2020. [Online]. Available: <https://arxiv.org/abs/2003.02139>
- [25] Y. Du, M.-H. Hsieh, T. Liu, and D. Tao, "Expressive power of parametrized quantum circuits," *Phys. Rev. Research*, vol. 2, p. 033125, Jul 2020. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevResearch.2.033125>
- [26] G. Cheng, X. Xie, J. Han, L. Guo, and G.-S. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 3735–3756, 2020.
- [27] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proceedings of the IEEE*, vol. 105, no. 10, pp. 1865–1883, 2017.
- [28] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, S. Ahmed, V. Ajith, M. S. Alam, G. Alonso-Linaje, B. AkashNarayanan, A. Asadi, J. M. Arrazola, U. Azad, S. Banning, C. Blank, T. R. Bromley, B. A. Cordier, J. Ceroni, A. Delgado, O. Di Matteo, A. Dusko, T. Garg, D. Guala, A. Hayes, R. Hill, A. Ijaz, T. Isacsson, D. Ittah, S. Jahangiri, P. Jain, E. Jiang, A. Khandelwal, K. Kottmann, R. A. Lang, C. Lee, T. Loke, A. Lowe, K. McKiernan, J. J. Meyer, J. A. Montañez-Barrera, R. Moyard, Z. Niu, L. J. O'Riordan, S. Oud, A. Panigrahi, C.-Y. Park, D. Polatajko, N. Quesada, C. Roberts, N. Sá, I. Schoch, B. Shi, S. Shu, S. Sim, A. Singh, I. Strandberg, J. Soni, A. Száva, S. Thabet, R. A. Vargas-Hernández, T. Vincent, N. Vitucci, M. Weber, D. Wierichs, R. Wiersema, M. Willmann, V. Wong, S. Zhang, and N. Killoran, "PennyLane: Automatic differentiation of hybrid quantum-classical computations," 2018. [Online]. Available: <https://arxiv.org/abs/1811.04968>
- [29] J. J. Meyer, "Fisher information in noisy intermediate-scale quantum applications," *Quantum*, vol. 5, p. 539, Sep 2021. [Online]. Available: <https://doi.org/10.22331/q-2021-09-09-539>
- [30] A. Ly, M. Marsman, J. Verhagen, R. Grasman, and E.-J. Wagenmakers, "A tutorial on Fisher information," 2017. [Online]. Available: <https://arxiv.org/abs/1705.01064>
- [31] S. Otgonbaatar and M. Datcu, "Assembly of a coreset of earth observation images on a small quantum computer," *Electronics*, vol. 10, no. 20, 2021. [Online]. Available: <https://www.mdpi.com/2079-9292/10/20/2482>
- [32] H.-Y. Huang, M. Broughton, M. Mohseni, R. Babbush, S. Boixo, H. Neven, and J. R. McClean, "Power of data in quantum machine learning," *Nature Communications*, vol. 12, no. 1, p. 2631, May 2021. [Online]. Available: <https://doi.org/10.1038/s41467-021-22539-9>
- [33] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nature Communications*, vol. 9, no. 1, p. 4812, Nov 2018. [Online]. Available: <https://doi.org/10.1038/s41467-018-07090-4>
- [34] S. Otgonbaatar and D. Kranzlmüller, "Exploiting the quantum advantage for satellite image processing: Quantum resource estimation," 2023.
- [35] I. Glasser, R. Sweke, N. Pancotti, J. Eisert, and J. I. Cirac, "Expressive power of tensor-network factorizations for probabilistic modeling, with applications from hidden markov models to quantum machine learning," 2019.

4.1.3 Publication C: Natural Embedding of the Stokes Parameters of Polarimetric Synthetic Aperture Radar Images in a Gate-based Quantum Computer

S. Otgonbaatar and M. Datcu

Abstract. We stated that **QML** models are designed to process quantum data (qubits) by leveraging both parameterized and non-parameterized quantum gates arranged by following a set of instructions such that they exhibit quantum advantage over conventional **ML** models. They promise a potential quantum advantage over traditional **ML** ones when their inputs are quantum data. However, real-world problems involve classical data (e.g., any digital number) that do not inherit the “specificity” of qubits. Hence, in the **publication A** and **publication B**, we designed and invented **QML** models by proposing two-level encoding and data re-uploading methods for embedding classical data in qubits such that they become more expressive, and hopefully outperform conventional classical learning models. Instead, this **publication C** proposes to train a **QML** model on a unique real-world dataset, so-called **PolSAR** images (see Chapter 2.5.1). Interestingly, **PolSAR** images are the doppelgänger of qubits or inherit the specificity of qubits. Our experiments demonstrate that the depth-one parameterized quantum circuit with a single input qubit learns to distinguish the Stokes parameters, which convey the physical scattering information of **PolSAR** images. Physical scattering information comprises single-bounced, double-bounced, and volume-scattered beams. More importantly, we note that we train our **QML** model via a hybrid classical-quantum approach on the C-band **PolSAR** image (covering the wavelength range from 1,530 nm to 1,565 nm) and test on an L-band **PolSAR** image (covering the wavelength range from 1,565 nm to 1,625 nm). This experiment validates that our **QML** model generalizes to different distributions having out-of-distribution properties when we embed wisely the underlying physical information of classical datasets in the input qubits.

Natural Embedding of the Stokes Parameters of Polarimetric Synthetic Aperture Radar Images in a Gate-Based Quantum Computer

Soronzonbold Otgonbaatar^{id} and Mihai Datcu^{id}, *Fellow, IEEE*

Abstract—Quantum algorithms are designed to process quantum data (quantum bits) in a gate-based quantum computer. They are proven rigorously that they reveal quantum advantages over conventional algorithms when their inputs are certain quantum data or some classical data mapped to quantum data. However, in a practical domain, data are classical in nature, and they are very big in dimension, size, and so on. Hence, there is a challenge to map (embed) classical data to quantum data, and even no quantum advantages of quantum algorithms are demonstrated over conventional ones when one processes the mapped classical data in a gate-based quantum computer. For the practical domain of earth observation (EO), due to the different sensors on remote-sensing platforms, we can map directly some types of EO data to quantum data. In particular, we have polarimetric synthetic aperture radar (PolSAR) images characterized by polarized beams. A polarized state of the polarized beam and a quantum bit are the Doppelgänger of a physical state. We map them to each other, and we name this direct mapping a *natural embedding*, otherwise an *artificial embedding*. Furthermore, we process our *naturally embedded* data in a gate-based quantum computer by using a quantum algorithm regardless of its quantum advantages over conventional techniques; namely, we use the QML network as a quantum algorithm to prove that we *naturally embedded* our data in input qubits of a gate-based quantum computer. Therefore, we employed and directly processed PolSAR images in a QML network. Furthermore, we designed and provided a QML network with an additional layer of a neural network, namely, a hybrid quantum-classical network, and demonstrate how to program (via optimization and backpropagation) this hybrid quantum-classical network when employing and processing PolSAR images. In this work, we used a gate-based quantum computer offered by an IBM Quantum and a classical simulator for a gate-based quantum computer. Our contribution is that we provided very specific EO data with a *natural embedding* feature, the Doppelgänger of quantum bits, and processed them in a hybrid quantum-classical network. More importantly, in the future, these PolSAR data can be processed by future quantum algorithms and future quantum computing platforms to obtain (or demonstrate) some quantum advantages over conventional techniques for EO problems.

Index Terms—Natural embedding, parameterized quantum circuit, polarimetric synthetic aperture radar (PolSAR), quantum machine learning (QML).

I. INTRODUCTION

RECENT breakthroughs in building a gate-based quantum computer with very few quantum bits (qubits) [1]

and in applying machine learning (ML) techniques to any annotated datasets led to a quantum algorithm called quantum ML (QML) being considered as a promising disruptive technique for a particular class of supervised learning methods [2]–[5], [6]. A quantum algorithm is an algorithm being processed in quantum computers, and a QML network is the network of *parameterized quantum gates* in a gate-based quantum computer. There is growing interest to apply a QML network to classical data [7], [8]. However, the gate-based quantum computer itself is posing several new challenges, for instance, how to map classical data to qubits (quantum data) depending on the limited number of its input qubits, or how to use the specificity of the “qubits” to obtain quantum advantages over nonquantum computing techniques, while ubiquitous data in practical domains are of classical nature. In particular, the input data play an important role in a quantum algorithm to obtain quantum advantages, and for example, in scientific studies [9], [10], their authors implied that QML networks achieve quantum advantages over a conventional technique only if classical data are *naturally embedded* in their input qubits, or their input data are quantum data.

Some studies proposed to embed classical data (e.g., RGB images) in quantum data by exploiting a conventional deep neural network (DNN) for its dimensionality reduction [11], [12]. Such an embedding procedure of classical data is named an *artificial embedding*, otherwise a *natural embedding*. Moreover, their QML network classifies *artificially embedded* data in qubits with no physical information even if the qubits carry physical information.

In this article, we introduce and provide a classical remotely sensed dataset with a *natural embedding* feature; in particular, we use polarimetric synthetic aperture radar (PolSAR) images of earth observation (EO) [13]–[15]. Then, we use the QML network as a quantum algorithm for processing our PolSAR images to prove that we *naturally embedded* our PolSAR data in qubits. The PolSAR images are obtained by using a PolSAR imaging technique. The PolSAR technique measures a scattering matrix S related to the incident and reflected Jones vector on the ground scene. The scattering matrix S preserves the physical scattering properties of the polarization-changing targets (e.g., *water*, *urban area*, and *vegetation*) [see Fig. 1(a)] [16]–[18].

More importantly, the PolSAR image can be represented by a number of Stokes parameters by assuming incident Jones vectors [19]. In this article, we represent the targets in a given PolSAR image by five Stokes parameters when assuming five

Manuscript received July 15, 2021; accepted August 31, 2021. Date of publication September 15, 2021; date of current version January 21, 2022. (Corresponding authors: Soronzonbold Otgonbaatar; Mihai Datcu.)

Soronzonbold Otgonbaatar is with German Aerospace Center (DLR), 82234 Weßling, Germany (e-mail: soronzonbold.otgonbaatar@dlr.de).

Mihai Datcu is with German Aerospace Center (DLR), 82234 Weßling, Germany, and also with Politehnica University of Bucharest (UPB), 060042 Bucharest, Romania (e-mail: mihai.datcu@dlr.de).

Digital Object Identifier 10.1109/TGRS.2021.3110056

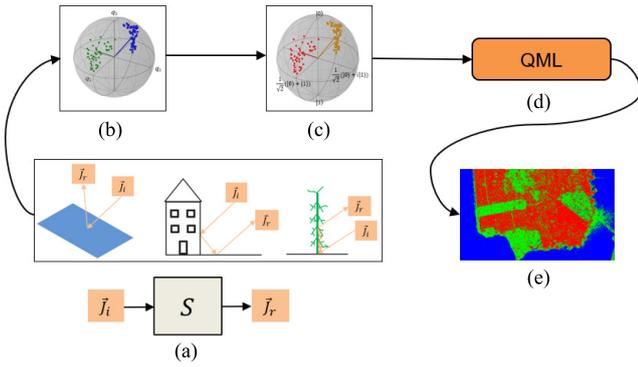


Fig. 1. This article in a pictorial representation. (a) Targets in a PolSAR image have different physical scattering properties for an incident Jones vector (a free variable). Given the incident Jones vector and given any polarization-changing target of a PolSAR image, we obtain a reflected Jones vector; $\vec{J}_i \xrightarrow{S} \vec{J}_r$, where S stands for the polarization-changing target/pixel of a given PolSAR image. (b) Stokes parameters. (c) Qubits. (d) QML network. (e) Outputs of the QML network that yields information about the polarization-changing targets in a PolSAR image.

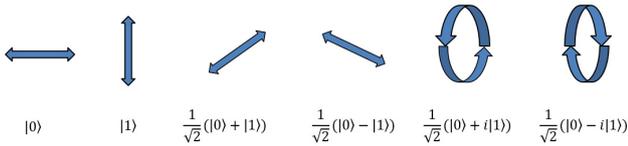


Fig. 2. Doppelganger role of Jones vectors and qubits. (Top) Jones vectors. (Left to Right) Horizontal, vertical, diagonal-up and -down, and left- and right-circular polarized states. (Bottom) Corresponding qubits.

different incident Jones vectors. Furthermore, the Jones vector is the Doppelganger of qubits (see Fig. 2), and the Stokes parameters then have one-to-one correspondences with the qubits; a qubit (or a two-state qubit) $|\psi\rangle \in \mathbb{C}^2$, $|\psi\rangle \in \{|0\rangle, |1\rangle\}$ is the quantum version of classical bits, and they can exist in superposition. This one-to-one correspondence property allows us to employ and process the PolSAR images as the input data of a QML network. Thus, we first *naturally* embed the Stokes parameters in qubits [see Fig. 1(b) and (c)]. Second, we demonstrate how to program (by optimization and back-propagation) the QML network and a hybrid quantum-classical network when employing and processing PolSAR images as its input data [see Fig. 1(d) and (e)]. The hybrid quantum-classical network is a learning network where the output of the QML network is connected to a layer of neurons. As real-world data, we use a C-band PolSAR image of *San Francisco*, USA, obtained from Radarsat-2, and L-band PolSAR image of *Oberpfaffenhofen*, Germany, taken by ESAR-L airborne.

This work is structured as follows. In Section II, we first provide some background on the Stokes parameters of our PolSAR images, and second, we derive the Stokes parameters of PolSAR images *San Francisco* and *Oberpfaffenhofen* in Section III and then propose a QML network as a hybrid quantum-classical network (see Section IV). Finally, we demonstrate how to program a hybrid quantum-classical

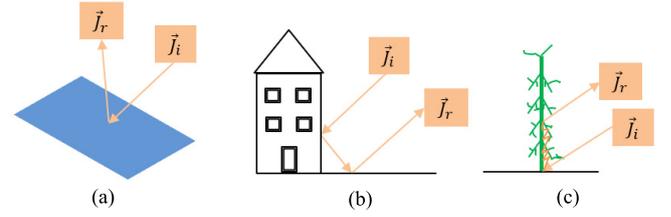


Fig. 3. Physical scattering properties of polarization-changing targets. (a) Single-bounced beam on water surface. (b) Double-bounced beam on corners of some targets. (c) Volume-scattered beam in trees and a medium.

network when applying synthetic quantum data as its input, and we validate that the Stokes parameters convey information about polarization-changing targets in PolSAR images by training them on a hybrid quantum-classical network (see Sections V and VI). Finally, we draw a conclusion in Section VII.

II. STOKES PARAMETERS OF POLSAR IMAGES DERIVED BY POLSAR TECHNIQUES

A. Introduction to the PolSAR Technique

A PolSAR system measures a 2×2 scattering matrix S of a beam characterizing targets in a ground scene at a given incident angle. The scattering matrix of a PolSAR image can be expressed by

$$S = \begin{pmatrix} s_{HH} & s_{HV} \\ s_{VH} & s_{VV} \end{pmatrix} \quad (1)$$

where each element of S is a complex-valued number; the first index of an element s_{ij} , $i, j \in \{H, V\}$ represents the polarization state of the incident polarized beam, and the second index represents the polarization state of the reflected polarized beam on targets [16], [17]. This scattering matrix preserves information on polarization-changing ground targets (see Figs. 1 and 3). The off-diagonal elements of S are equal $s_{VH} = s_{HV}$ when PolSAR images (e.g., *San Francisco* and *Oberpfaffenhofen*) are fully polarized PolSAR images obtained by a monostatic radar.

Furthermore, the ground targets imagined in a PolSAR image have disparate physical scattering properties, e.g., geometrical structures, scattering each incident polarized beam from a PolSAR system differently. For example, an incident polarized beam on a rough surface (water) has the scattering properties of a single-bounced beam, the incident polarized beam on some corners of buildings has the scattering properties of a double-bounced beam, and the incident polarized beam on trees has the scattering properties of a volume-scattered beam (see Fig. 3). These different scattering properties of any targets are represented by a so-called Pauli vector. The Pauli vector can be written as

$$\begin{aligned} \mathbf{k} &= \frac{1}{\sqrt{2}} (s_{HH} + s_{VV} \quad s_{HH} - s_{VV} \quad 2s_{HV})^T \\ &= \frac{1}{\sqrt{2}} (k_1 \quad k_2 \quad k_3)^T \end{aligned} \quad (2)$$

where subscript T represents transposition, and k_1, k_2 , and k_3 represent a single-bounced beam, a double-bounced beam, and a volume-scattered beam, respectively [16], [17], [20].

B. Derivation of Stokes Parameters

An incident/reflected beam on the ground scene can be expressed by

$$\vec{E} = \vec{E}_0 \exp i\phi \quad (3)$$

where ϕ is a phase, and \vec{E}_0 is a complex amplitude vector of an incident/reflected beam. A complex amplitude vector can be expressed in a polarization basis $\{H, V\}$ by

$$\vec{E}_0 = E_{H0}\hat{H} + E_{V0}\hat{V}. \quad (4)$$

This complex amplitude vector can be rewritten as

$$\vec{J} = \begin{pmatrix} E_{H0} \\ E_{V0} \end{pmatrix} = \begin{pmatrix} |E_{H0}|e^{i\phi_H} \\ |E_{V0}|e^{i\phi_V} \end{pmatrix} \quad (5)$$

where ϕ_i are the phases of the polarized states. This expression is called a *Jones vector* \vec{J} .

Moreover, the 2×2 scattering matrix expressed by (1) is a mapping of an incident Jones vector such that

$$S: \vec{J}_i \rightarrow \vec{J}_r, \quad \vec{J}_r = S\vec{J}_i \quad (6)$$

where \vec{J}_i is an incident Jones vector and \vec{J}_r is a reflected Jones vector. More importantly, the incident Jones vector is a free variable that one can manipulate. In matrix form, (6) can be rewritten as

$$\begin{pmatrix} E_{H0}^r \\ E_{V0}^r \end{pmatrix} = \begin{pmatrix} s_{HH} & s_{HV} \\ s_{VH} & s_{VV} \end{pmatrix} \begin{pmatrix} E_{H0}^i \\ E_{V0}^i \end{pmatrix}. \quad (7)$$

The coherency matrix (intensity) of the reflected Jones vector \vec{J}_r is defined by

$$J = \begin{pmatrix} \langle E_{H0}^r E_{H0}^{r*} \rangle & \langle E_{H0}^r E_{V0}^{r*} \rangle \\ \langle E_{V0}^r E_{H0}^{r*} \rangle & \langle E_{V0}^r E_{V0}^{r*} \rangle \end{pmatrix} = \begin{pmatrix} J_{HH} & J_{HV} \\ J_{VH} & J_{VV} \end{pmatrix} \quad (8)$$

where $\langle \cdot \rangle$ stands for spatial averaging and $*$ for conjugation. This expression is called a Jones coherency matrix of a reflected Jones vector. Furthermore, we reexpress this coherency matrix by

$$\begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} |E_{H0}|^2 + |E_{V0}|^2 \\ |E_{H0}|^2 - |E_{V0}|^2 \\ 2|E_{H0}||E_{V0}|\cos\phi_{HV} \\ 2|E_{H0}||E_{V0}|\sin\phi_{HV} \end{pmatrix} = \begin{pmatrix} J_{HH} + J_{VV} \\ J_{HH} - J_{VV} \\ J_{VH} + J_{HV} \\ i(J_{HV} - J_{VH}) \end{pmatrix} \quad (9)$$

where q_1 , q_2 , and q_3 are *Stokes vectors*; $\phi_{HV} = \phi_H - \phi_V$ is the phase difference in radians with $\phi_H = \arg(E_{H0})$, and $\phi_V = \arg(E_{V0})$. In addition, we normalize these *Stokes vectors* such that

$$q_1 = \frac{q_1}{q_0}, \quad q_2 = \frac{q_2}{q_0}, \quad q_3 = \frac{q_3}{q_0} \quad (10)$$

and the normalized q_1 , q_2 , and q_3 are called *Stokes parameters*.

III. STOKES PARAMETERS OF OUR POLSAR IMAGES

As real-world PolSAR images, we use the PolSAR images of *San Francisco* and *Oberpfaffenhofen* [see Fig. 4 (top)]. For any PolSAR image, its polarization-changing targets S can be represented by any numbers of Stokes parameters based on the incident Jones vector (a free variable).

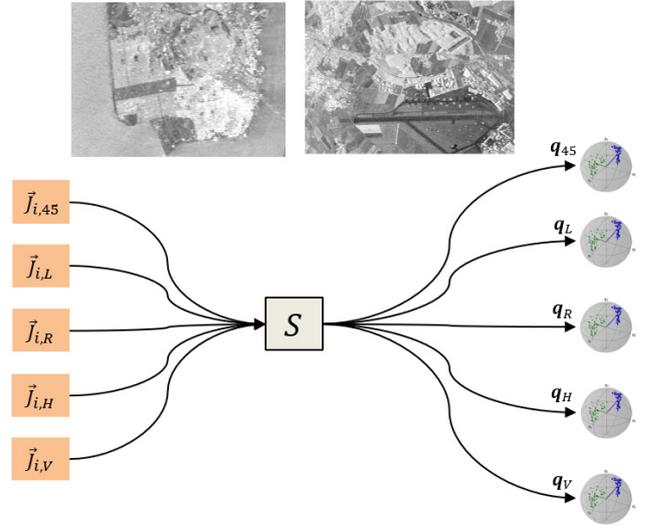


Fig. 4. Polarization-changing target S in a PolSAR image is represented by five Stokes parameters (features). (Top) PolSAR images of *San Francisco* and *Oberpfaffenhofen*. (Bottom; Left to Right) Incident Jones vectors, a polarization-changing target S , and the corresponding Stokes parameters.

In this article, we assume the incident Jones vector $\vec{J}_i \in \{\vec{J}_{i,45}, \vec{J}_{i,L}, \vec{J}_{i,R}, \vec{J}_{i,H}, \vec{J}_{i,V}\}$ shown in Fig. 4 (bottom)

$$\begin{aligned} \vec{J}_{i,45} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix}, & \vec{J}_{i,L} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ i \end{pmatrix}, & \vec{J}_{i,R} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -i \end{pmatrix} \\ \vec{J}_{i,V} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 0 \\ 1 \end{pmatrix}, & \vec{J}_{i,H} &= \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 0 \end{pmatrix} \end{aligned} \quad (11)$$

where $\vec{J}_{i,45}$ is the polarized state along a 45° diagonal, $\vec{J}_{i,L}$ is the polarized state of the left-handed circular beam, $\vec{J}_{i,R}$ is the polarized state of the right-handed circular beam, $\vec{J}_{i,V}$ is the vertically polarized state, and $\vec{J}_{i,H}$ is the horizontally polarized state [21].

Then, we obtain each target of our PolSAR images the corresponding Stokes parameters by using (8)–(10)

$$\mathbf{q} = \{q_{45}, q_L, q_R, q_V, q_H\}, \quad \mathbf{q}_i \in \mathbb{R}^3. \quad (12)$$

We characterize each target pixel by five Stokes parameters (features) by manipulating the incident Jones vector. More importantly, we have five times more features for each polarization-changing target in a PolSAR image than in an original image (see Fig. 4). We use these Stokes parameters as inputs of a hybrid quantum-classical network, as shown in step I of Fig. 5.

In Sections IV and V, we design a hybrid quantum-classical network, and we demonstrate how a hybrid quantum-classical network is programmed (using optimization and backpropagation) to update its weights when applying a synthetic quantum dataset as its input. For PolSAR images, we embed the Stokes parameters in the qubits of a hybrid quantum-classical network, as shown in step II of Fig. 5. Finally, we demonstrate the effectiveness of using PolSAR images in a hybrid quantum-classical network by recognizing the target classes of PolSAR images, namely, *San Francisco* and *Oberpfaffenhofen* (see step III of Fig. 5).

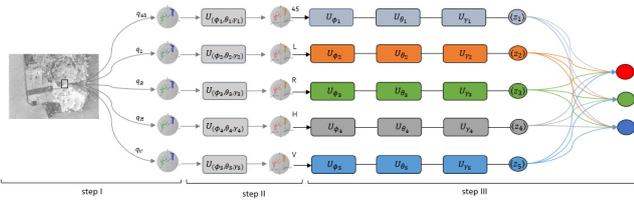


Fig. 5. General steps for a hybrid quantum-classical network for our PolSAR image, e.g., *San Francisco*: (step-I) Stokes parameters as inputs. (step-II) embedding of Stokes parameters in qubits. (step-III) hybrid quantum-classical network for analyzing PolSAR images.

IV. HYBRID QUANTUM-CLASSICAL NETWORK

A QML network consists of *two-state qubits* and *parameterized quantum gates*. A *two-state qubit* is a complex vector in a Hilbert space denoted as a ket vector $|\psi\rangle \in \{|0\rangle, |1\rangle\}$

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad (13)$$

and as a bra vector $\langle\psi| = |\psi\rangle^\dagger$, where \dagger represents both transpose and conjugation. The two-state qubits can result in superposition as

$$|\psi\rangle = c_1|0\rangle + c_2|1\rangle \quad \text{s.t.} \quad |c_1|^2 + |c_2|^2 = 1 \quad (14)$$

where c_1 and c_2 are complex numbers, and $P_i = |c_i|^2$ represents the probability for obtaining a state $|i\rangle$ after a measurement (see Fig. 6). Namely, in quantum physics, by measurements, we obtain the eigenvalue of an observable \hat{z} with a certain probability P_i .

Parameterized quantum gates are unitary operators defined by

$$\hat{U} = \hat{U}(\gamma, \theta, \phi) = \hat{U}_\gamma \hat{U}_\theta \hat{U}_\phi \quad (15)$$

where γ , θ , and ϕ are parameters (angles), and they can be written in matrix form as

$$\hat{U}_\gamma = \begin{pmatrix} e^{i\gamma} & 0 \\ 0 & e^{-i\gamma} \end{pmatrix}, \quad \hat{U}_\theta = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \quad (16)$$

$$\hat{U}_\phi = \begin{pmatrix} e^{i\phi} & 0 \\ 0 & e^{-i\phi} \end{pmatrix}. \quad (17)$$

These unitary operators induce rotations of the qubits on/in the Bloch sphere [21] such that $|\psi'\rangle = \hat{U}|\psi\rangle$, as shown in Fig. 6.

For a QML network, we measure the expectation value of an operator \hat{z} , and the operator \hat{z} is decomposed into its eigenvalues z_i and eigenvectors $|z_i\rangle$ as follows:

$$\hat{z} = \sum_i z_i |z_i\rangle\langle z_i|, \quad z_i \in \{-1, +1\}. \quad (18)$$

Then, the expectation value of an operator \hat{z} is defined by

$$\langle z \rangle = \langle \psi | \hat{z} | \psi \rangle = \sum_i P_i z_i, \quad \langle z \rangle \in [-1, +1] \quad (19)$$

where P_i is the probability of obtaining the eigenvalue z_i given a quantum state $|\psi\rangle$.

Furthermore, we added a conventional layer of neurons to the outputs of a QML network. Such a network is called a

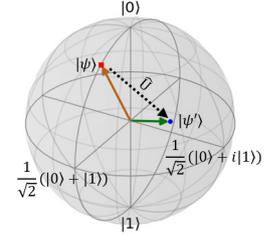


Fig. 6. Unitary transformation of a qubit into another qubit on the Bloch sphere. $\hat{U}:|\psi\rangle \rightarrow |\psi'\rangle$, where $|\psi\rangle = c_1|0\rangle + c_2|1\rangle$ and $\hat{U}|\psi\rangle = |\psi'\rangle = c'_1|0\rangle + c'_2|1\rangle$.

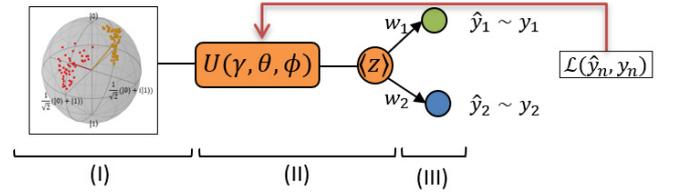


Fig. 7. Hybrid quantum-classical network for the classification of a quantum dataset: (I) Quantum dataset (qubits $|\psi\rangle$), (II) QML network, and (III) conventional layer of neurons, e.g., two neurons.

hybrid quantum-classical network, as shown in Fig. 7. For simplicity, we designed a hybrid quantum-classical network by using only three *parameterized quantum gates* expressed in (15) and a layer of two neurons for a binary quantum dataset. We used these unitary operators both for the embedding of classical data in qubits and for designing a hybrid quantum-classical network.

For programming the hybrid quantum-classical network, we expressed our expectation value by

$$\langle z \rangle = \langle \psi | U^\dagger(\gamma, \theta, \phi) \hat{z} U(\gamma, \theta, \phi) | \psi \rangle. \quad (20)$$

In addition, we assumed the above expectation value as a function

$$z(\theta_1, \theta_2, \theta_3) = \langle z \rangle \quad (21)$$

where $\theta_1 = \gamma$, $\theta_2 = \theta$, $\theta_3 = \phi$, and the outputs of the neurons are then expressed such that

$$\hat{y}_n = A(b_n + w_n z(\theta_1, \theta_2, \theta_3)) \quad (22)$$

where $A = A(\cdot)$ is a nonlinear activation function, \hat{y}_n is its predicted output, b_n is its bias, and w_n is its edge parameter.

1) Optimization and Backpropagation: We have five parameters $\vec{w} = (w_1, w_2, \theta_1, \theta_2, \theta_3)$, and we optimize a total loss function $\mathcal{L} = \mathcal{L}(\hat{y}_n, y_n)$; namely, we update and learn the parameters as follows:

$$\vec{w}_{\text{new}} = \vec{w}_{\text{old}} - \lambda \nabla \cdot \mathcal{L} \quad (23)$$

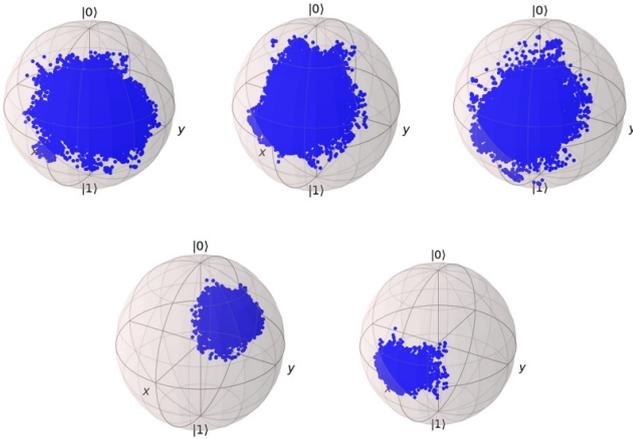


Fig. 8. For *San Francisco*, its embedded Stokes parameters in qubits. Each Stokes parameter carries different information of a polarization-changing target. (Top Left) Qubits with \mathbf{q}_{45} . (Top Middle) Qubits with \mathbf{q}_L . (Top Right) Qubits with \mathbf{q}_R . (Bottom Left) Qubits with \mathbf{q}_V . (Bottom Right) Qubits with \mathbf{q}_H .

where λ is the learning rate, and by backpropagation, we obtain

$$\begin{aligned} \nabla \cdot \mathcal{L} &= (\partial \mathcal{L} / \partial w_1, \partial \mathcal{L} / \partial w_2, \partial \mathcal{L} / \partial \theta_1, \partial \mathcal{L} / \partial \theta_2, \partial \mathcal{L} / \partial \theta_3)^T \\ \frac{\partial \mathcal{L}}{\partial w_n} &= \frac{\partial \mathcal{L}}{\partial \hat{y}_n} \frac{\partial \hat{y}_n}{\partial w_n}, \quad n = 1, 2 \\ \frac{\partial \mathcal{L}}{\partial \theta_i} &= \sum_{n=1}^2 \frac{\partial \mathcal{L}}{\partial \hat{y}_n} \frac{\partial \hat{y}_n}{\partial z(\theta_1, \theta_2, \theta_3)} \frac{\partial z(\theta_1, \theta_2, \theta_3)}{\partial \theta_i}, \quad i = 1, 2, 3 \end{aligned} \quad (24)$$

which we compute on a conventional computer.

V. PROCESSING OF A SYNTHETIC QUANTUM DATASET ON A HYBRID QUANTUM-CLASSICAL NETWORK

As a proof of concept, we analyzed a small set of a synthetic quantum dataset, and we used the hybrid quantum-classical network shown in Fig. 7. It has the advantage that we could easily understand the implementation and the performance of a hybrid quantum-classical network.

In our case, the synthetic quantum dataset consists of the points $(|\psi\rangle_n, y_n)$, $n = 1, 2, \dots, 100$, where its first half corresponds to a class $y_n = (1, 0)$ represented in yellow in Fig. 7, and its other half corresponds to a class $y_n = (0, 1)$ represented in red in Fig. 7. We generated this quantum dataset according to

$$|\psi\rangle_n = \cos\left(\frac{\alpha}{2}\right)|0\rangle_n + e^{i\beta} \sin\left(\frac{\alpha}{2}\right)|1\rangle_n \quad (25)$$

where we assumed $\alpha = 0.6$ for a class label $y_n = (0, 1)$ and $\alpha = 3$ for a class label $y_n = (1, 0)$. Hence, we did not need the embedding procedure.

For the implementation of the hybrid quantum-classical network, we employed an IBM quantum computer (an *ibmq-armonk* qubit) and a classical simulator for a quantum computer; here, the *ibmq-armonk* qubit of an IBM quantum computer is a noisy qubit, while the qubit of a classical simulator is a perfect one. With both of these quantum computers



Fig. 9. Visualization of *San Francisco*. (Left) Google map of *San Francisco* exhibiting the classes *urban area*, *vegetation*, and *sea water*. (Middle) Red contour presents a part of the *urban area* with a different orientation (see the red arrow); namely, a *rotated urban area*. (Right) Visualization of the Pauli vectors of *San Francisco* expressed by (2). The Pauli vector clearly distinguishes the *rotated urban area* from the remainder of the *urban area*. Blue: *sea water* (a single-bounced beam). Pink: *urban area* (a double-bounced beam). Light green: *rotated urban area* (inside the red contour) with a different orientation from the remainder (a volume-scattered beam). Dark green: *vegetation* (a volume-scattered beam).

and a classical simulator, we reached a classification *accuracy* and *loss* of (1.000, 0.0083) for our synthetic quantum dataset. This finding leads to the very important conclusion that for a noisy quantum computer, our hybrid network does not need any error correction processing for a noisy qubit. More importantly, the classical layer of our hybrid network corrects a classification mistake made by a noisy qubit.

The insights gained from the classification of our synthetic quantum dataset on the hybrid quantum-classical network are given as follows.

- 1) A classical dataset needs to be *naturally* embedded in qubits due to the very small number of qubits.
- 2) A hybrid quantum-classical network does not need any error correction processing for a noisy qubit. This could be proven by the accuracy of our experiment on a synthetic quantum dataset.
- 3) A classical layer handles the mistakes made by a noisy qubit.
- 4) The publicly available IBM quantum computer has a very high overhead.

VI. EMBEDDING AND PROCESSING OF POLSAR IMAGES IN A HYBRID QUANTUM-CLASSICAL NETWORK

A. Embedding the Stokes Parameters of PolSAR Images in Qubits

We used the PolSAR images of *San Francisco* and *Oberpfaffenhofen* for processing in a hybrid quantum-classical network, and we represented them by the Stokes parameters $\mathbf{q} = (\mathbf{q}_{45}, \mathbf{q}_L, \mathbf{q}_R, \mathbf{q}_V, \mathbf{q}_H)$ presented in Section III. Here, we embedded the Stokes parameters in the qubits of a hybrid quantum-classical network by applying (15). The Stokes parameters of *San Francisco* and *Oberpfaffenhofen* are naturally equivalent to qubits due to the Doppelganger role of reflected Jones vectors and qubits. The advantage is that we do not need to reduce our PolSAR images in their given spatial dimensionality or train another QML network to embed them in qubits. Thus, we embedded the Stokes parameters in qubits in a two-step procedure.

- 1) A single-qubit is prepared in a state $|0\rangle$.
- 2) The Stokes parameters \mathbf{q}_i for each pixel of our PolSAR images are embedded in qubits by using (15) such that

$$U(\gamma_1, \theta_2, \phi_3) : |0\rangle \rightarrow |\psi\rangle, \quad |\psi\rangle = U(\gamma_1, \theta_2, \phi_3)|0\rangle$$

where $\gamma_1 = q_1$, $\theta_2 = q_2$, and $\phi_3 = q_3$ are elements of each \mathbf{q}_i , $i = 45, L, R, V, H$; the state $|\psi\rangle$ sits in/on a Bloch sphere.

For the case of the PolSAR image of *San Francisco*, we show its embedded Stokes parameters carrying different scattering properties for targets in Fig. 8.

B. Processing PolSAR Images in a Hybrid Quantum-Classical Network

We processed the PolSAR images *San Francisco* and *Oberpfaffenhofen* by exploiting their five Stokes parameters in a hybrid quantum-classical network. In particular, we analyzed that whether the Stokes parameters of a given PolSAR image carry some physical scattering properties of polarization-changing targets by recognizing them in a hybrid quantum-classical network. The physical scattering properties are a *single-bounced beam*, a *double-bounced beam*, and a *volume-scattered beam* on polarization-changing targets (see Figs. 3 and 9). Here, we use the PolSAR image of *San Francisco* as a training and a validation dataset and the PolSAR image of *Oberpfaffenhofen* as a test dataset.

For the training dataset, we selected four classes of *San Francisco* that are *urban area* (its physical scattering property: a double-bounced beam), *rotated urban area* (its physical scattering property: a volume-scattered beam), *sea water* (its physical scattering property: a single-bounced beam), and *vegetation* (its physical scattering property: a volume-scattered beam) shown in Fig. 9 [22], [23]; each class image has a size of 590×497 pixels. Once having defined our training dataset, we designed our experiment by the following procedures (see step-I and step-III in Fig. 5).

- 1) Derive the five Stokes parameters from our training dataset.
- 2) Embed these Stokes parameters in qubits by Steps 1) and 2) of Section VI-A.
- 3) Design/train a hybrid quantum-classical network.
- 4) Employ an *Adam* optimizer and the categorical cross entropy for the training process.

Then, we ran our experiment for the recognition of targets in *San Francisco* in three different scenarios.

A *first scenario for the binary classes urban area and rotated urban area* is given as follows.

- 1) We represented these binary classes only by the Stokes parameter \mathbf{q}_{45} .
- 2) We embedded \mathbf{q}_{45} in qubits and trained the qubits by employing the hybrid quantum-classical network shown in Fig. 10.
- 3) The validation accuracy reached 0.8005.

In this first scenario, we represented the binary classes of *urban area* and *rotated urban area* of *San Francisco* by the Stokes parameter \mathbf{q}_{45} . Then, we trained and distinguished them by applying the hybrid quantum-classical network shown in Fig. 10. Finally, we compared our detected results with respect to their Pauli vectors (see Fig. 11). This comparison confirms that the Stokes parameter \mathbf{q}_{45} describes the physical scattering properties of *urban area* and *rotated urban area*.

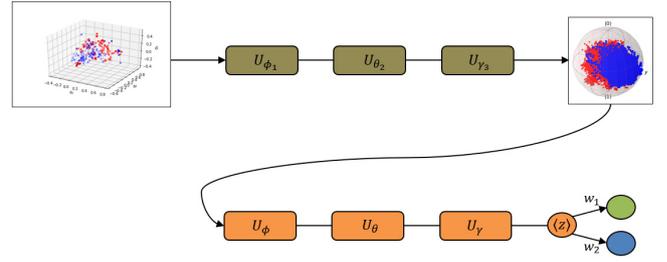


Fig. 10. Our hybrid quantum-classical network for the embedding and the training of our binary classes of *San Francisco* with \mathbf{q}_{45} . (Top) Embedding of our Stokes parameter \mathbf{q}_{45} in qubits. (Bottom) hybrid quantum-classical network for training these qubits.

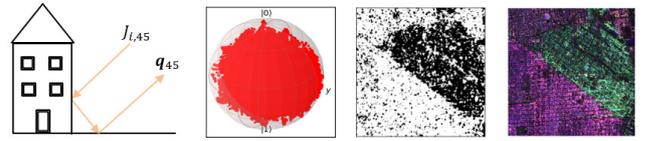


Fig. 11. (Left to Right) *urban area* and the *rotated urban area* of *San Francisco* are characterized by the Stokes parameter \mathbf{q}_{45} , the embedded Stokes parameters in qubits, the visual results for the binary classification (*urban area* and *rotated urban area*) in our hybrid quantum-classical network shown in Fig. 10, and the Pauli vector of *urban area* and *rotated urban area*.

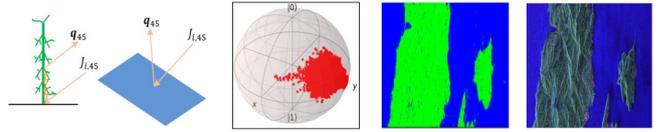


Fig. 12. (Left to Right) *Sea water* and *vegetation* of *San Francisco* are characterized by the Stokes parameter \mathbf{q}_{45} , the embedded Stokes parameters in qubits, the visual results for the binary classification (*sea water* and *vegetation*) in our hybrid quantum-classical network shown in Fig. 10, and the Pauli vector of *sea water* and *vegetation*.

A *second scenario for the binary class sea water and vegetation* is given as follows.

- 1) We represented this binary class only by the Stokes parameter \mathbf{q}_{45} .
- 2) We embedded \mathbf{q}_{45} in qubits and trained the qubits by employing the hybrid quantum-classical network shown in Fig. 10.
- 3) The validation accuracy reached 0.9613.

In this second scenario, we represented the binary class of *sea water* and *vegetation* by the Stokes parameter \mathbf{q}_{45} . Then, we trained and recognized the classes by applying the hybrid quantum-classical network shown in Fig. 10 as in the previous scenario. We also benchmarked our findings with respect to their Pauli vectors; the visual (target recognition) result and their Pauli vector are shown in Fig. 12. This visual result again proves that the Stokes parameter \mathbf{q}_{45} characterizes the physical scattering properties of *sea water* and *vegetation*.

A *final scenario for the three classes, vegetation, urban area, and sea water*, is given as follows.

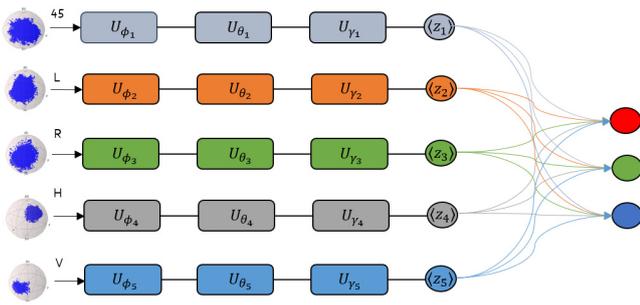


Fig. 13. Hybrid quantum-classical network for the training of three classes of *San Francisco* with five simultaneous inputs of \mathbf{q}_i . The hybrid quantum-classical network for the training of given qubits (the embedding of Stokes parameters in qubits is not shown).

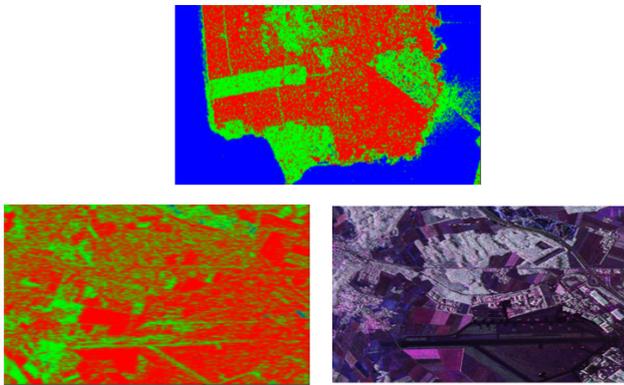


Fig. 14. Recognition of targets in PolSAR images (*blue*: a single-bounced beam, *red*: a double-bounced beam, and *green*: a volume-scattered beam). (Top) Target recognition in the PolSAR image of *San Francisco*. (Bottom Left) Targets recognition in the PolSAR image of *Oberpfaffenhofen*. (Right) Pauli vector of the PolSAR image of *Oberpfaffenhofen*.

- 1) We represented these three classes by the Stokes parameters $\mathbf{q} = (\mathbf{q}_{45}, \mathbf{q}_L, \mathbf{q}_R, \mathbf{q}_V, \mathbf{q}_H)$.
- 2) We embedded \mathbf{q} in qubits and trained the three classes by applying the hybrid quantum-classical network shown in Fig. 13.
- 3) The validation accuracy reached 0.8362.

In this final scenario, we represented each target of our PolSAR image by five Stokes parameters $\mathbf{q} = (\mathbf{q}_{45}, \mathbf{q}_L, \mathbf{q}_R, \mathbf{q}_V, \mathbf{q}_H)$, and we trained them by using the hybrid quantum-classical network shown in Fig. 13. We visualize the identified results in Fig. 14 Top. The visual results demonstrate that our network identified *vegetation* and *rotated urban area* as the same class due to the same physical scattering property, namely, a volume-scattered beam [see Fig. 9 (right)]. This result conclusively proves that the Stokes parameters contain the physical scattering properties of targets.

Furthermore, we even tested our trained hybrid quantum-classical network on *Oberpfaffenhofen* and verified our test result with respect to its Pauli vector. The visual results explicitly show that our network recognized the physical scattering properties of targets in the PolSAR image of *Oberpfaffenhofen* [see Fig. 14 (bottom)].

VII. CONCLUSION

The inputs of quantum algorithms are quantum data, and for practical applications, there is a persisting challenge to map classical data to quantum data. Hence, we provided real-world EO measurement data that inherit quantum nature; in particular, PolSAR images. The PolSAR images consist of polarization-changing targets that are characterized by the reflected Jones vectors. The reflected Jones vectors are the Doppelgänger of qubits. Therefore, we mapped the reflected Jones vectors (the Stokes parameters) of PolSAR images to quantum bits, and we designed and provided hybrid quantum-classical networks as a quantum algorithm to prove that we *naturally* embedded our data in qubits. Then, we investigated whether the Stokes parameters convey the physical scattering properties of the targets of PolSAR images by recognizing them in our hybrid quantum-classical networks. In this work, we represented the targets of PolSAR images by five Stokes parameters. As real-world PolSAR images, we used two types of PolSAR images, namely, *San Francisco* and *Oberpfaffenhofen*.

In more detail, we provided PolSAR images with a *natural embedding* feature and investigated whether their Stokes parameters convey the physical scattering properties of the PolSAR images by recognizing them in the hybrid quantum-classical networks. We processed the PolSAR image of *San Francisco* in three different scenarios of its Stokes parameters by using two different hybrid quantum-classical networks. After training these hybrid quantum-classical networks in different scenarios for the Stokes parameters of *San Francisco*, we tested these trained hybrid quantum-classical networks on the PolSAR image of *Oberpfaffenhofen*. Our obtained results demonstrate that the Stokes parameters convey the physical scattering properties of PolSAR images. In addition, we ran our hybrid quantum-classical network both on an IBM quantum computer and a classical simulator of a quantum computer. The classical layer of our hybrid quantum-classical network corrects an error made by a noisy qubit (*ibmq-armonk*) of the IBM quantum computer; the IBM quantum computer that is publicly available has a high overhead.

In terms of our ongoing and future work, we study deeply a hybrid quantum-classical network and future quantum computing platforms for processing PolSAR images due to the Doppelgänger role of the Jones vectors and qubits. Furthermore, we design a hybrid quantum-classical network for distinguishing targets with the same physical scattering properties in PolSAR images.

ACKNOWLEDGMENT

The authors would like to greatly acknowledge Gottfried Schwarz (DLR, Oberpfaffenhofen) for his valuable comments and contributions for enhancing the quality of this article.

REFERENCES

- [1] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, p. 79, Aug. 2018, doi: 10.22331/q-2018-08-06-79.

- [2] S. Otgonbaatar and M. Datcu, "A quantum annealer for subset feature selection and the classification of hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 7057–7065, Jul. 2021. [Online]. Available: <https://ieeexplore.ieee.org/document/9477115/>
- [3] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, Sep. 2017, doi: [10.1038/nature23474](https://doi.org/10.1038/nature23474).
- [4] V. Dunjko and H. J. Briegel, "Machine learning & artificial intelligence in the quantum domain: A review of recent progress," *Rep. Prog. Phys.*, vol. 81, no. 7, Jun. 2018, Art. no. 074001, doi: [10.1088/1361-6633/aab406](https://doi.org/10.1088/1361-6633/aab406).
- [5] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Phys. Rev. Lett.*, vol. 113, Sep. 2014, Art. no. 130503, doi: [10.1103/PhysRevLett.113.130503](https://doi.org/10.1103/PhysRevLett.113.130503).
- [6] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Phys.*, vol. 10, pp. 631–633, Jul. 2014, doi: [10.1038/nphys3029](https://doi.org/10.1038/nphys3029).
- [7] N. H. Nguyen, E. C. Behrman, M. A. Moustafa, and J. E. Steck, "Benchmarking neural networks for quantum computations," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 7, pp. 2522–2531, Jul. 2020.
- [8] Y. Li, M. Tian, G. Liu, C. Peng, and L. Jiao, "Quantum optimization and quantum learning: A survey," *IEEE Access*, vol. 8, pp. 23568–23593, Jan. 2020.
- [9] H.-Y. Huang *et al.*, "Power of data in quantum machine learning," *Nature Commun.*, vol. 12, no. 1, pp. 1–9, May 2021, doi: [10.1038/s41467-021-22539-9](https://doi.org/10.1038/s41467-021-22539-9).
- [10] J. M. Kübler, S. Buchholz, and B. Schölkopf, "The inductive bias of quantum kernels," 2021, [arXiv:2106.03747](https://arxiv.org/abs/2106.03747). [Online]. Available: <http://arxiv.org/abs/2106.03747>
- [11] M. Schuld and N. Killoran, "Quantum machine learning in feature Hilbert spaces," *Phys. Rev. Lett.*, vol. 122, Feb. 2019, Art. no. 040504, doi: [10.1103/PhysRevLett.122.040504](https://doi.org/10.1103/PhysRevLett.122.040504).
- [12] S. Lloyd, M. Schuld, A. Ijaz, J. Izaac, and N. Killoran, "Quantum embeddings for machine learning," May 2020, [arXiv:2001.03622](https://arxiv.org/abs/2001.03622). [Online]. Available: <http://arxiv.org/abs/2001.03622>
- [13] S. R. Cloude and K. P. Papathanassiou, "Polarimetric SAR interferometry," *IEEE Trans. Geosci. Remote Sens.*, vol. 36, no. 5, pp. 1551–1565, Sep. 1998.
- [14] S. R. Cloude and E. Pottier, "An entropy based classification scheme for land applications of polarimetric SAR," *IEEE Trans. Geosci. Remote Sens.*, vol. 35, no. 1, pp. 68–78, Jan. 1997.
- [15] A. G. Mullissa, C. Persello, and A. Stein, "PolSARNet: A deep fully convolutional network for polarimetric SAR image classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 12, pp. 5300–5309, Dec. 2019.
- [16] J. S. Lee *et al.*, "A review of polarimetric SAR algorithms and their applications," *J. Photogram. Remote Sens.*, vol. 9, no. 3, pp. 31–80, 2004.
- [17] A. Moreira, P. Prats-Iraola, M. Younis, G. Krieger, I. Hajnsek, and K. P. Papathanassiou, "A tutorial on synthetic aperture radar," *IEEE Geosci. Remote Sens. Mag.*, vol. 1, no. 1, pp. 6–43, Mar. 2013.
- [18] H. Dong, B. Zou, L. Zhang, and S. Zhang, "Automatic design of CNNs via differentiable neural architecture search for PolSAR image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 9, pp. 6362–6375, Sep. 2020.
- [19] F. Shang and A. Hirose, "Quaternion neural-network-based PolSAR land classification in Poincare-sphere-parameter space," *IEEE Trans. Geosci. Remote Sens.*, vol. 52, no. 9, pp. 5693–5703, Sep. 2014.
- [20] G. Singh, S. Mohanty, Y. Yamazaki, and Y. Yamaguchi, "Physical scattering interpretation of POLSAR coherency matrix by using compound scattering phenomenon," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 4, pp. 2541–2556, Apr. 2020.
- [21] S. Barnett, Ed., *Quantum Information*. London, U.K.: Oxford Univ. Press, 2009.
- [22] D. Ratha, A. Bhattacharya, and A. C. Frery, "Unsupervised classification of PolSAR data using a scattering similarity measure derived from a geodesic distance," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 1, pp. 151–155, Jan. 2018.
- [23] S. Dey, A. Bhattacharya, D. Ratha, D. Mandal, and A. C. Frery, "Target characterization and scattering power decomposition for full and compact polarimetric SAR data," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 3981–3998, May 2021.



Soronzonbold Otgonbaatar received the B.Sc. degree from the National University of Mongolia, Ulaanbaatar, Mongolia, in 2014, and the M.Sc. degree in theoretical particle physics from the University of Siegen, Siegen, Germany, in 2016. He is currently pursuing the Ph.D. degree with German Aerospace Center, Weßling, Germany.

His research interests include computational science, artificial intelligence (i.e., machine learning), quantum computing and algorithms, and quantum machine learning with practical applications in remote sensing data.



Mihai Datcu (Fellow, IEEE) received the M.S. and Ph.D. degrees in electronics and telecommunications from the University Politehnica of Bucharest (UPB), Bucharest, Romania, in 1978 and 1986, respectively, and the Habilitation a Diriger Des Recherches degree in computer science from Louis Pasteur University, Strasbourg, France, in 1999.

From 1992 to 2002, he had a longer Invited Professor Assignment with the Swiss Federal Institute of Technology (ETH Zürich), Zürich, Switzerland.

He was a Visiting Professor with the University of Oviedo, Oviedo, Spain; Louis Pasteur University; International Space University, Strasbourg; the University of Siegen, Siegen, Germany; the University of Innsbruck, Innsbruck, Austria; the University of Alcalá, Alcalá de Henares, Spain; Tor Vergata University, Rome, Italy; the Universidad Pontificia de Salamanca, Madrid, Spain; the University of Camerino, Camerino, Italy; and the Swiss Center for Scientific Computing, Manno, Switzerland. Since 1981, he has been a Professor with the Department of Applied Electronics and Information Engineering, Faculty of Electronics, Telecommunications and Information Technology, UPB. Since 1993, he has been a Scientist with the German Aerospace Center (DLR), Weßling, Germany. He is developing algorithms for model-based information retrieval from high-complexity signals and methods for scene understanding from very-high-resolution synthetic aperture radar (SAR) and interferometric SAR data. Since 2011, he has been leading the Immersive Visual Information Mining Research Laboratory, Munich Aerospace Faculty, and he is the Director of the Research Center for Spatial Information, UPB. Since 2001, he has been initiating and leading the Competence Center on Information Extraction and Image Understanding for Earth Observation, ParisTech, Paris Institute of Technology, Télécom Paris, Paris, France, a collaboration of DLR with the French Space Agency (CNES). He has been a Professor with the DLR-CNES Chair, ParisTech, Paris Institute of Technology, Télécom Paris. He has initiated the European frame of projects for image information mining (IIM) and is involved in research programs for information extraction, data mining and knowledge discovery, and data understanding with the European Space Agency (ESA), Paris, NASA, and in a variety of national and European projects. He and his team have developed and are developing the operational IIM processor in the Payload Ground Segment systems for the German missions, TerraSAR-X, TanDEM-X, and the ESA Sentinel-1 and Sentinel-2. He is currently a Senior Scientist and the Data Intelligence and Knowledge Discovery Research Group Leader with the Remote Sensing Technology Institute, DLR, and involved in the DLR-ONERA Joint Virtual Center for AI in Aerospace. He is also a Visiting Professor with the ESA's Φ -Lab. His research interests include explainable and physics-aware artificial intelligence, smart sensors design, and quantum machine learning with applications in earth observation.

Dr. Datcu is also a member of the ESA Working Group Big Data from Space. He was a recipient of the Best Paper Award and the IEEE Geoscience and Remote Sensing Society Prize in 2006, the National Order of Merit with the Rank of Knight, for outstanding international research results, awarded by the President of Romania, in 2008, and the Romanian Academy Prize Traian Vuia for the development of the SAADI image analysis system and his activity in image processing in 1987. He was also a recipient of the Chaire d'excellence internationale Blaise Pascal 2017 for international recognition in the field of data science in earth observation. He has served as a Co-Organizer for international conferences and workshops and a Guest Editor for special issues on AI, quantum machine learning, or big data of the IEEE and other journals.

4.2 A D-Wave Quantum Annealer for Earth Observation

We study and benchmark optimization problems deployed on a **QA** device when using **EO** datasets. We present a technique to map classical optimization problems to **QUBO** forms and demonstrate the potential benefits and disadvantages of using **QA** approaches over a conventional simulated annealing approach. This chapter is based on the following publications:

- **D:** S. Otgonbaatar and M. Datcu, “A Quantum Annealer for Network Flow Minimization in InSAR Images,” EUSAR 2021; *13th European Conference on Synthetic Aperture Radar*, Leipzig, Germany, 2021, pp. 1-4 (Virtual Symposium) [68].
- **E:** S. Otgonbaatar and M. Datcu, “A Quantum Annealer for Subset Feature Selection and the Classification of Hyperspectral Images,” in *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 7057-7065, 2021, doi: 10.1109/JSTARS.2021.3095377 [64].
- **F:** S. Otgonbaatar and M. Datcu, “Assembly of a Coreset of Earth Observation Images on a Small Quantum Computer,” *Electronics* 10, no. 20: 2482., 2021, doi: electronics10202482 [65].

4.2.1 Publication D: A Quantum Annealer for Network Flow Minimization in InSAR Images

S. Otgonbaatar and M. Datcu

Abstract. A quantum annealer is a novel computing technology designed to solve QUBO problems more efficiently than classical annealers and conventional methods. This validates the study and programming of the D-Wave quantum annealer for real-world quadratic optimization problems. This study explores how to program the D-Wave quantum annealer for real-world optimization problems. Specifically, we focus on the QP-to-QUBO formulation, which is explained in Chapter 3.2.1 of this dissertation, and how the parameters of the D-Wave quantum annealer affect the quality of its solutions. We then examine several InSAR applications in EO and identify a residue connection problem in the phase-unwrapping procedure that is posed as a network flow minimization problem. We use the InSAR image of Cape Verde volcano as a real-world problem. To compute the network flow minimization problem on the D-Wave quantum annealer, we transform it into a corresponding QUBO problem and ran several experiments by setting the parameters of the D-Wave quantum annealer in different configurations, following the workflow presented in Chapter 3.2.1 of this dissertation. We study the effect of the D-Wave quantum annealer's parameters on finding a solution to the network flow minimization problem. However, due to the large number of variables involved in the network flow minimization problem, our obtained solutions were not convincing. This is because they cannot be embedded efficiently in the limited number and connectivity of the input qubits of the D-Wave quantum annealer. Nonetheless, this study provides valuable insights into choosing and executing proper real-world problems on the D-Wave quantum annealer and other QA devices.

Quantum annealer for network flow minimization in InSAR images

Soronzonbold Otgonbaatar¹ and Mihai Datcu, Fellow IEEE²

^{1,2}German Aerospace Center (DLR), Oberpfaffenhofen, Germany

²University POLITEHNICA of Bucharest (UPB), Romania

Abstract

Quantum Annealer (QA) is well-suited for a certain class of optimization problems which can be expressed as a Quadratic Unconstrained Binary Optimization (QUBO) problem. A QUBO problem belongs to the family of Integer Programming problems which are called the NP-hard optimization problems. Feasible solutions of such problems can be found by using classical optimization techniques. However, studies claim that QA can find a feasible global solution that is faster than a classical annealer for QUBO problems. Hence, it appears promising to program and use the QA-to-QUBO approach for Earth Observation. In search of the QUBO problem in the domain of Earth Observation, we examined several Interferometric Synthetic Aperture Radar (InSAR) applications and identified a residue connection problem in the phase unwrapping procedure. In particular, we consider the residue connection problem with multiples of 2π as a QUBO problem. For this practical problem, we studied how to formulate this QUBO problem, and we examined the challenges to program the D-Wave quantum annealer, in particular, embedding the QUBO problem into our QA architecture with a so-called Pegasus topology, and the annealing parameter settings in the D-Wave quantum annealer. We then analysed the parameter effects on finding the global minimum of the residue connection problem. From these results, we derived and enhanced our insight for programming future quantum annealers; for instance, choosing real-world problems in Earth Observation, conceiving the embedding procedure, and the tuning of the annealing parameters.

1 Introduction

Quantum Annealer is a novel computing technology promising to provide solutions for hard-optimization or sampling problems. A specific type of QA is the D-Wave quantum annealer which has a specific graph topology called *Pegasus* and *Chimera* [1]. The vertices of these graph topologies represent the physical variables, and the edges represent the interaction strengths among these physical variables. However, not all vertices are connected to all others through the edges; in other words, the interaction among the physical variables is restricted. Further, D-Wave quantum annealer promises to solve only a QUBO-like problem, and the variables of a QUBO problem are logical variables. To solve a QUBO problem via the D-Wave quantum annealer, we need to program the D-Wave quantum annealer; the programming consists of two steps: *embedding* and the setting of the *annealing parameters*. The *embedding* is a procedure to map the QUBO problem to the D-Wave quantum annealer topology, namely, our Pegasus topology; in particular, mapping (embedding) the logical variables to the corresponding physical variables. After embedding the QUBO problem, the D-Wave quantum annealer uses a meta-heuristic (quantum annealing) process for finding a global minimum solution of the QUBO problem; the quantum annealing process is parametrized by its *annealing parameter* [2]. Hence, the *embedding* procedure and the *annealing parameter* setting affect the problem solutions. Several studies already dealt with the embedding of QUBO problems to the physical variables of Chimera graphs, dealing with both academic [3]-[6] as well as with

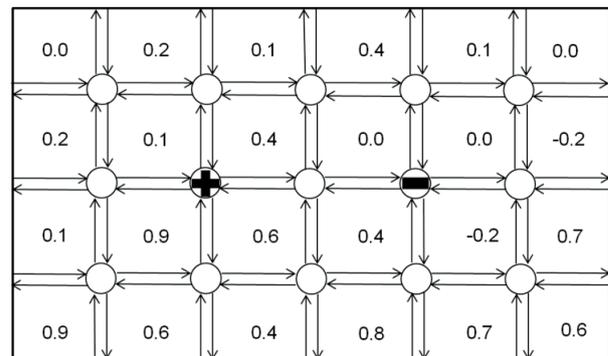


Figure 1 An example of 2D wrapped phase values and their residues. Empty vertices have zero 0 residues. Phases of 2π are normalized to 1.

real-world applications, [7], [8]. There are even indications that D-Wave quantum annealer may solve certain QUBO problems faster than classical methods by exploiting quantum effects such as tunnelling and entanglement [9].

The QUBO problems are well-known in mathematics, physics, and industrial optimization problems due to the complexity of obtaining a good global optimum. In the field of mathematics, [3], this author identified NP-hard problems and formulated them as QUBO problems to find their global minimum solution by exploiting QA. In the field of graph theory, [6], other author selected certain graph problems $G = (V, E)$ to find their solution by using the quantum annealing process on QA. Such graph problems were to find minimum cuts and maximum cliques of

the graph. In the field of the high energy physics, [10], the wrapped measurement data can be modelled as a QUBO problem, and a quantum annealing process is then employed for un-wrapping the data on a D-Wave quantum annealer. These diverse research findings motivated us to this work. However, so far, no advantage for real-world optimization problems could be demonstrated.

Therefore, this work aims to investigate to programming of a D-Wave quantum annealer for QUBO problems in Earth Observation. As a real-world problem in Earth Observation, we chose the residue connection problem with multiples of 2π in the phase unwrapping procedure and posed this problem as the QUBO problem. The residue connection problem with multiples of 2π is solvable with several classical optimization techniques, for instance, Network Flow Minimization (NFM) algorithm, [11], [12], in which their authors contributed and used well-known optimization algorithms for finding the global minimum of the objective function for the residue connection (Fig. 1). We even can pose this problem simply and easily as a QUBO problem. Hence, it validates to employ the quantum annealing process on a D-Wave quantum annealer.

Therefore, in this paper, we exhibit a way to formulate the objective function of the residue connection problem as a QUBO problem, and we then embed this QUBO problem to the Pegasus topology. Finally, we run the quantum annealing process with different settings of the annealing parameters to assess the performance of the D-Wave quantum annealer.

1.1 Basics of the residues in the phase unwrapping procedure

Synthetic Aperture Radar (SAR) is a coherent imaging system operating in the microwave domain. Interferometric SAR (InSAR) techniques exploit the information of the phases of two complex-valued SAR signals acquired from different positions in order to construct an ‘‘Interferogram’’; hence, the phases of two complex-valued SAR signals are called the *interferometric phases* which allow us to estimate the height of the observed scene [13], [14]. The interferometric phases of two complex-valued signals are measured, and their differences are wrapped into the region $[-\pi, \pi]$ called *wrapped phase* [15].

For a 1D signal, the true or *unwrapped phases* are then obtained by means of adding multiples of 2π to the wrapped phase values if the absolute value of a wrapped phase difference (i.e., its gradient) is less than $-\pi$. For a 2D signal, instead, the same concept as for 1D signals does not provide a satisfactory result for the unwrapped phases due to signal noise, and hence, this concept would cause some error propagation for the unwrapped phases. The signal noises are identified as the *residues* which are results of integration of the wrapped phase gradients along 2×2 square grids (Fig. 1); the residues take values of $\{-1, 0, +1\}$. Hence, the wrapped phases of a 2D signal are non-conservative and path-dependent [11]. Furthermore, commonly used phase unwrapping procedures are based on the assumption that the wrapped phase gradients $\Delta\psi(n)$ known experimentally are equal to unwrapped

phase gradients $\Delta\phi(n)$ at most of the instances, and according to this assumption, the unwrapped phase gradients can be reconstructed from

$$\Delta\phi(n) = \Delta\psi(n) + 2\pi\Delta k(n), \quad (1)$$

where $\Delta\phi(n), \Delta\psi(n) \in [-\pi, +\pi]$ with n representing a discretized location of phase values on the 2D plane, and $k(n) \in \mathbb{N}$ are integer variables. If the integer variables $k(n)$ are known then we can integrate Eq. (1) to reconstruct the unwrapped phases. To avoid residues to be enclosed in the integration path of the wrapped phases, the residues need to be connected, and a number of the connected residues has to be neutral to ensure the path invariance along the integration path. Otherwise, an imbalanced number of the residues distorts the path integral of the wrapped phase gradients (Fig. 1). Hence, the residue connection problem can be expressed as a minimization problem,

$$\min_{\{x_1^+, x_1^-, x_2^+, x_2^-\}} \left[\sum_{i=0}^{N-2} \sum_{j=0}^{M-1} c_1(i, j) [x_1^+(i, j) + x_1^-(i, j)] + \sum_{i=0}^{N-1} \sum_{j=0}^{M-2} c_2(i, j) [x_2^+(i, j) + x_2^-(i, j)] \right], \quad (2)$$

such that

$$\begin{aligned} & x_1^+(i, j+1) - x_1^-(i, j+1) - x_1^+(i, j) + x_1^-(i, j) \\ & - x_2^+(i+1, j) + x_2^-(i+1, j) + x_2^+(i, j) - x_2^-(i, j) \\ & = -\frac{1}{2\pi} (\psi_1(i, j+1) - \psi_1(i, j) - \psi_2(i+1, j) + \psi_2(i, j)), \\ & \bar{x}_1^+ \geq 0, \bar{x}_1^- \geq 0, \bar{x}_1^+, \bar{x}_1^- \in \mathbb{N}, \\ & \bar{x}_2^+ \geq 0, \bar{x}_2^- \geq 0, \bar{x}_2^+, \bar{x}_2^- \in \mathbb{N}. \end{aligned} \quad (3)$$

$x_1^+(i, j), x_1^-(i, j)$ are the flow along vertices $(i-1, j) \leftrightarrow (i, j)$, $x_2^+(i, j), x_2^-(i, j)$ are the flow along vertices $(i, j-1) \leftrightarrow (i, j)$, and $c_1(i, j)$, and $c_2(i, j)$ are the unit cost of the flows (Fig. 1). The global minimum of this objective function can be found efficiently by using the NFM algorithm [11], [12].

1.2 The QUBO problem formulation

We consider less densely populated residues with multiples of 2π such that the $\{x_1^+, x_1^-, x_2^+, x_2^-\}$'s of the objective function in Eq. (2) take the binary values $\{0, 1\}$; here, the residues with multiples of 2π mean that the integer variables only take the values one or zero while, for the residues with more than multiples of 2π , the integer variables can be any integer numbers. For either case, we can express the objective function, Eq. (2), as the QUBO problem;

$$H(\mathbf{x}) = \sum_{i,j} x_i Q_{ij} x_j, \quad (4)$$

where the logical variables $\mathbf{x} \in \{0, 1\}^n$, and Q_{ij} includes a bias term h_i and the interaction strength of the logical variables J_{ij} .

For the residue connection problem with multiples of 2π , we can easily embed it into the Pegasus topology and assess the performance of the D-Wave quantum annealer with respect to a conventional algorithm. In this case, each residue value, $b(i, j) \in \{-1, 0, +1\}$, must be equal to the total edge value, namely the conservation of flow at the vertex $b(i, j)$ (Fig. 1). The constraint Eq. (3) then remains as before

$$\begin{aligned} & \underbrace{[x_1^+(i, j+1) - x_1^-(i, j+1)] + [-x_1^+(i, j) + x_1^-(i, j)]}_{x_1} \\ & + \underbrace{[-x_2^+(i+1, j) + x_2^-(i+1, j)] + [x_2^+(i, j) - x_2^-(i, j)]}_{x_2} \\ & = b(i, j), \quad x_1^\pm, x_2^\pm \in \{0, +1\}, \end{aligned} \quad (5)$$

here $b(i, j) = -\frac{1}{2\pi}(\psi_1(i, j+1) - \psi_1(i, j) - \psi_2(i+1, j) + \psi_2(i, j))$. Further, we can formulate the QUBO form for the residue connection problem by means of expressing Eq. (2) and Eq. (5) in a quadratic form as

$$\begin{aligned} & \min_{\{x_1^+, x_1^-, x_2^+, x_2^-\}} \left[\sum_{i=0}^{N-2} \sum_{j=0}^{M-1} c_1(i, j)[x_1^+(i, j) + x_1^-(i, j)] \right. \\ & + \sum_{i=0}^{N-1} \sum_{j=0}^{M-2} c_2(i, j)[x_2^+(i, j) + x_2^-(i, j)] \\ & \left. + \lambda \sum_{(i, j) \in S} [x_1 + x_2 - b]^2 \right], \end{aligned} \quad (6)$$

where we suppose that $\lambda = 1$ and

$$\begin{aligned} & \sum_{(i, j) \in S} [x_1 + x_2 - b]^2 = \\ & = \sum_{(i, j) \in S} 2x_1x_2 - 2b(x_1 + x_2) + x_1 + x_2. \end{aligned}$$

Eq. (6), is already in the form of the QUBO problem described by Eq. (4), and in the end, we have the QUBO formulation for the residue connection problem with multiples of 2π ; in particular, the residues have the multiples of 2π .

Furthermore, we can derive the QUBO problem with the residues having more than multiples of 2π by using the fixed-bit encoding of the integer variable $x_i^* \in \{x_1^+, x_1^-, x_2^+, x_2^-\}$'s,

$$x_i^* = \sum_{k=0}^{n-1} 2^k q_{n \times i + k}^*, \quad x_i^* \in \mathbb{N}, q_{n \times i + k}^* \in \{0, +1\}, \quad (7)$$

where n stands for the number of bits, and "*" represents "+" or "-".

However, we consider only the QUBO problem of Eq. (6) for a real-world data set to examine the performance of the D-Wave quantum annealer; in particular, *Cape Verde*

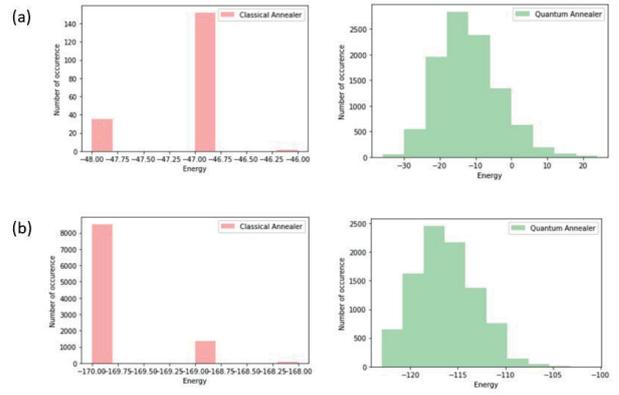


Figure 2 The minimum results of the QUBO problem (minimum energy) on the classical annealer and the D-Wave quantum annealer. (a) Minimum energies for 3×3 and (b) 7×5 residue problems.

Volcano, Africa, as a real-world Sentinel-1 data set. In the next section, we embed the QUBO problem for *Cape Verde Volcano* into the Pegasus topology and run the quantum annealing process with different annealing parameters on the D-Wave quantum annealer.

2 The D-Wave quantum annealer and our experiment

The company D-Wave systems offers a specific kind of a quantum annealer. This quantum annealer, called D-Wave quantum annealer, anneals two-state quantum particles to find their most probable state. The most probable state is equivalent to minimizing the QUBO problem defined by Eq. (4). However, the interaction strength J_{ij} is characterized by the specific topology *Pegasus* on the D-Wave quantum annealer. Hence, the QUBO formulation for a practical problem needs to be embedded into the Pegasus topology. When addressing the QUBO formulation for *Cape Verde Volcano* contained in Sentinel-1 data, we considered the two cases, namely 3×3 and 7×5 residues, such that

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix}. \quad (8)$$

Here, the number of the local variables of the QUBO problem, given by Eq. (6), is "48" for the first case, while the second case uses "164". We attempted to find the global solution of the QUBO problem for both cases on a classical annealer and the D-Wave quantum annealer. Then we benchmarked the results of the D-Wave quantum annealer with respect to the classical annealer. To program the D-Wave quantum annealer for obtaining the global solution of Eq. (6), we followed a two-step approach:

1. *Embedding*: Embedding of the logical variables into

their corresponding physical variables (in Pegasus topology). In this first step, we used a method named *minor-embedding* offered by the D-Wave quantum annealer. We represented a single logical variable by at most five physical variables.

2. *Annealing parameters*: These are the annealing time, the number of reads, and the chain strength. The annealing time is the time to run the quantum annealing process. We set this time to a value ranging from $20\mu s$ to $190\mu s$, the number of reads to a value from 1000 to 10000. Additionally, the chain strength is the interaction strength which was set to 50. We even tried a method called *reverse annealing* which starts the quantum annealing process at the known local solution of the problem of interest.

We obtained the results for the both residue cases given above by programming the D-Wave quantum annealer, and we have validated these results by inserting them into the constraint of Eq. (5). Our results demonstrate that, in the first case of the 3×3 residues, the D-Wave quantum annealer finds a better solution than the classical annealer. However, in the latter case of the 7×5 residues, the classical annealer finds a better solution than the D-Wave quantum annealer. We plotted the possible minima of the QUBO problem (minimum energies) for both cases of (8), as a demonstration (Fig. 2). For these scenarios, we can conclude that the D-Wave quantum annealer outperforms the classical annealer, while the number of the local variables in the QUBO problem is very limited.

3 Conclusion and Discussion

This work is a first attempt to enhance insights into a quantum annealer, and perceive some challenges to program the D-Wave quantum annealer for future Earth Observation quantum technology. To leverage the D-Wave quantum annealer to obtain a better solution of the optimization problem, we chose a well-known optimization problem in Earth observation, and mapped it to a QUBO problem; in particular, the residue connection problem of the phase unwrapping procedure in the InSAR.

For the chosen residue connection problem with multiples of 2π , we assessed and programmed the D-Wave quantum annealer with different annealing parameters. The results obtained from the D-Wave quantum annealer were benchmarked with respect to a classical annealer. It turned out that the D-Wave quantum annealer performs better than the selected classical annealer while the number of local variables is very small. In further studies, we will investigate other real-world problems of remote sensing and Earth Observation which shall benefit greatly from a QUBO formulation.

4 ACKNOWLEDGMENT

We would like to acknowledge Bogdan-Florin Florea (UPB) whose initial attempt to solve this problem, the

valuable suggestions of Gottfried Schwarz (DLR, Oberpfaffenhofen), and Tobias Stollenwerk (DLR, Koeln), for his comments on the first draft of this paper. The authors also gratefully acknowledge the Juelich Supercomputing Centre (<https://www.fzjuelich.de/ias/jsc>) for funding of this project by providing computing time through the Juelich UNified Infrastructure for Quantum computing (JUNIQU) on the D-Wave quantum annealer.

5 Literature

- [1] D-Wave ocean software.
- [2] E. Farhi, J. Goldstone, S. Gutmann, "A Quantum Approximate Optimization Algorithm", arXiv: 1411.4028.
- [3] L. Andrew, "Ising formulations of many NP problems", arXiv: 1302.5843, 2014.
- [4] W. Cruz-Santos, S.E. Venegas-Andraca, M. Lanzagorta, "A QUBO Formulation of Minimum Multicut Problem Instances in Trees for D-Wave Quantum Annealers", *Sci Rep*, vol. 9, 17216, 2019.
- [5] K. Zick, O. Shehab, M. French, "Experimental QA: case study involving the graph isomorphism problem", *Sci Rep*, vol. 5, 11168, 2015.
- [6] G. Chapuis, H. Djidjev, G. Hahn, et al. "Finding Maximum Cliques on the D-Wave Quantum Annealer", *J Sign Process Syst*, vol. 91, pp. 363–377, 2019.
- [7] S. Otgonbaatar, M. Datcu, "Quantum Annealing Approach: Feature extraction and segmentation of Synthetic Aperture Radar Image", *IGARSS*, 2020.
- [8] T. Stollenwerk, V. Michaud, E. Lobe, et al., "Image Acquisition Planning for Earth Observation Satellites with a Quantum Annealer", arXiv: 2006.09724.
- [9] V. Denchev, S. Boixo, S. Isakov, et al., "What is the Computational Value of Finite-Range Tunneling?", *Phys. Rev. X*, vol. 6, 031015, 2016.
- [10] K. James, R. Cormier, R.D. Sipio, P. Wittek, "Unfolding as quantum annealing", *J. High Energ. Phys.*, 128, 2019.
- [11] M. Costantini, "A novel phase unwrapping method based on network programming," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36, pp. 813–821, 1998.
- [12] C.W. Chen, H.A. Zebker, "Network approaches to two-dimensional phase unwrapping: intractability and two new algorithms," *J. Opt. Soc. Am. A*, vol. 17, pp. 401–414, 2000.
- [13] H. Zebker, R. Goldstein, "Topographic mapping from interferometric SAR observations," *J. Geophys. Res.*, vol. 91, pp. 4993–4999, 1986.
- [14] A.K. Gabriel, R.M. Goldstein, H.A. Zebker, "Mapping small elevation changes over large areas: differential radar interferometry," *J. Geophys. Res.*, vol. 94, pp. 9183–9191, 1989.
- [15] H.A. Zebker, Y. Lu, "Phase unwrapping algorithms for radar interferometry: residue-cut, least-squares, and synthesis algorithms," *J. Opt. Soc. Am. A*, vol. 15, pp. 586–598, 1998.

4.2.2 Publication E: A Quantum Annealer for Subset Feature Selection and the Classification of Hyperspectral Images

S. Otgonbaatar and M. Datcu

Abstract. We develop a new experiment for quantum feature selection and quantum classifiers for satellite datasets. We utilize a Mutual Information (MI)-based technique to select highly informative subset features from large-scale satellite datasets, known as the QUBO-based subset feature selection. Additionally, we design novel multi-label quantum classifiers, Quantum boost (Qboost) and Qboost-Plus with the help of Error Correcting Output Code (ECOC). We deploy these quantum approaches on the D-Wave quantum annealer when using the Indian Pine hyperspectral image with around 200 contiguous spectral bands and 16 classes. As some spectral bands in the hyperspectral dataset are redundant and uninformative, selecting the most informative subset band becomes an NP-hard problem since there are 2^{200} combinations. Therefore, our experiment justifies the exploration of quantum approaches (feature selection and classification) for the Indian Pine hyperspectral image on the D-Wave quantum annealer. To implement our quantum approaches on the D-Wave quantum annealer, we strictly follow this dissertation's workflow described in Chapter 3.2.1. Furthermore, we compare the performance of our multi-label quantum boost classifiers with classical classifiers such as the decision tree, an SVM, and adaptive boost classifiers. Our experimental results demonstrate that our multi-label quantum classifier can generate correct multi-label images competitive with conventional methods following our QUBO-based subset feature selection method. The Qboost-Plus classifier with the help of ECOC executed on the D-Wave quantum annealer outperforms the adaptive boost classifier for generating multi-label image data points.

A Quantum Annealer for Subset Feature Selection and the Classification of Hyperspectral Images

Soronzonbold Otgonbaatar  and Mihai Datcu , *Fellow, IEEE*

Abstract—Hyperspectral images (HSIs) showing objects belonging to several distinct target classes are characterized by dozens of spectral bands being available. However, some of these spectral bands are redundant and/or noisy, and hence, selecting highly informative and trustworthy bands for each class is a vital step for classification and for saving internal storage space; then the selected bands are termed a highly informative spectral band subset. We use a mutual information (MI)-based method to select the spectral band subset of a given class and two additional binary quantum classifiers, namely a quantum boost (Qboost) and a quantum boost plus (Qboost-Plus) classifier, to classify a two-label dataset characterized by the selected band subset. We pose both our MI-based band subset selection problem and the binary quantum classifiers as a quadratic unconstrained binary optimization (QUBO) problem. Such a quadratic problem is solvable with the help of conventional optimization techniques. However, the QUBO problem is an NP-hard global optimization problem, and hence, it is worthwhile for applying a quantum annealer. Thus, we adapted our MI-based optimization problem for selecting highly informative bands for each class of a given HSI to be run on a D-Wave quantum annealer. After the selection of these highly informative bands for each class, we employ our binary quantum classifiers to a two-label dataset on the D-Wave quantum annealer. In addition, we provide a novel multilabel classifier exploiting an error-encoding output code when using our binary quantum classifiers. As a real-world dataset in Earth observation, we used the well-known AVIRIS HSI of Indian Pine, north-western Indiana, USA. We can demonstrate that the MI-based band subset selection problem can be run on a D-Wave quantum annealer that selects the highly informative spectral band subset for each target class in the Indian Pine HSI. We can also prove that our binary quantum classifiers and our novel multilabel classifier generate a correct two- and multilabel dataset characterized by their selected bands and with high accuracy such as having been produced by conventional classifiers—and even better in some instances.

Index Terms—D-wave quantum annealer (QA), feature selection, hyperspectral images (HSIs), mutual information (MI), quantum machine learning, quantum classifier.

Manuscript received March 4, 2021; revised May 21, 2021; accepted June 27, 2021. Date of publication July 7, 2021; date of current version July 23, 2021. This work was supported by the Juelich Supercomputing Centre (<https://www.fzjuelich.de/ias/jsc>) by providing computing time through the Juelich UNified Infrastructure for Quantum computing (JUNIQU) on a D-Wave QA. (Corresponding author: Soronzonbold Otgonbaatar.)

Soronzonbold Otgonbaatar is with the German Aerospace Center (DLR), 82234 Wessling, Germany, and also with the University of Siegen, 57076 Siegen, Germany (e-mail: soronzonbold.otgonbaatar@dlr.de).

Mihai Datcu is with the German Aerospace Center (DLR), 82234 Wessling, Germany, and also with University POLITEHNICA of Bucharest (UPB), 060042 Bucharest, Romania (e-mail: mihai.datcu@dlr.de).

Digital Object Identifier 10.1109/JSTARS.2021.3095377

I. INTRODUCTION

A QUANTUM annealer (QA) is a computing machine configured as a graph network $G = (E, V)$, at each vertex of which particles are residing, and its edges define the interaction strengths among these particles, which are in quantum states *ups* or *downs*. For a D-Wave QA, the graph G has a specific network topology named *Pegasus*, in which only certain edges are connected. In particular, the interaction among the particles is constrained [1], [2].

A D-Wave QA works as a metaheuristic process, which is dedicated to tackle specific classes of optimization problems, e.g., quadratic unconstrained binary optimization (QUBO) problems. There are theoretical studies that a D-Wave QA can solve these QUBO problems faster than a conventional annealer (even for NP problems) [3], [4]. However, currently, there are no indications of computational advantages for real-world problems. For practical applications, several studies are devoted to benchmark and assess a D-Wave QA for an operational planning and feature extraction from remotely sensed images [5], [6].

For a real-world dataset in Earth observation, remotely sensed images differ in their image content representations due to the diverse satellite platforms with their different types of sensors. When we want to use a D-Wave QA with an Earth observation dataset, some of the challenges are the proper choice of appropriate remotely sensed images specified by their image content representations such as their spatial information, polarization states, spectral bands, and the embedding of a given dataset in the topology of a D-Wave QA. Here, we consider hyperspectral images (HSIs), and a selection of their highly informative band subset is a very vital procedure in Earth observation. Hence, we use a mutual information (MI)-based optimization method to select the highly informative band subset, and more importantly, we can easily embed and optimize the MI-based optimization method in the *Pegasus* topology of a D-Wave QA. Therefore, HSIs are one of the most proper datasets in Earth observation for a D-Wave QA than others. In particular, HSIs became an important field of study to classify or identify objects in a ground scene such as roads, land cover, or agriculture since each object is characterized by a high-dimensional vector of the different spectral bands within the given full wavelength range. Due to the rich information content of the spectral bands, some of these bands carry more discriminatory information than others. Hence, some studies are focused on extracting highly informative features or a dimensionality reduction of HSIs, for instance, by using deep learning networks or principal component analysis

(PCA) [7], [8]. On the other hand, some researchers focused on how to select a highly informative band subset by using the concept of the information theory; in particular, MI-based methods that provide a measure of independence between several spectral bands. Moreover, these MI-based methods are based on prior or reference knowledge of the spectral signatures of objects; such knowledge can be obtained in specific spectral-signature databases of common ground targets [9], [10].

In this article, we use an Indian Pine HSI of Indian Pine composed of $l = 16$ distinct classes. Each class is characterized by $n = 200$ bands (features); thus, the number of subsets of the features of a given class, e.g., $y_S = 1$, is 2^n combinations. A way to find the best feature subset of this given class is to try all combinations. This is clearly computationally expensive for a large number of these features.

Hence, in the first part of this study, we introduce an MI-based subset feature selection problem as a global optimization problem for the Indian Pine HSI. Then, we propose to optimize this MI-based band subset selection problem on a D-Wave QA. As a first step, we map the MI-based band subset selection problem to a QUBO-based band subset selection problem. This is our first *problem mapping* step. Second, we optimize this mapping problem on a D-Wave QA; *quantum optimization* [11], [12]; this part was strongly motivated by a feature selection tutorial offered by D-Wave [2].

In the second part of this study, we use binary quantum classifiers, namely a quantum boost (Qboost) and a quantum boost plus (Qboost-Plus) classifier, in contrast to an adaptive boost (Adaboost) classifier [13], [14]. We first apply these quantum classifiers to a two-label dataset of the Indian Pine HSI, and second, we provide a novel multilabel classifier via an error-encoding output code (ECOC) when using our binary quantum classifiers [15], [16]; each resulting class is discriminated by the selected bands in the first part of our study. We also benchmarked and assessed these binary quantum classifiers and the novel multilabel classifier with respect to conventional classifiers, a decision tree classifier (DTC), a support vector machine (SVM), and an Adaboost classifier.

Our contribution in this article is then an attempt to benchmark and assess a D-Wave QA for Earth observation data and to recognize the challenges that are encountered with real-world datasets and future QAs or devices. Toward these goals, we are employing a D-Wave QA for feature selection and classification of the Indian Pine HSI as a machine learning technique; our contribution consists of a three-step approach.

- 1) Feature selection on a D-Wave QA: The MI-based band subset selection.
- 2) Binary classification on a D-Wave QA: The binary quantum classifiers to a two-label dataset characterized by those selected bands.
- 3) Multi-label classification on a D-Wave QA: The ECOC generates a multilabel dataset when we are using our binary quantum classifiers.

Moreover, the D-Wave QA may prove relevant even if we are not intending to demonstrate its advantage over a conventional annealer.

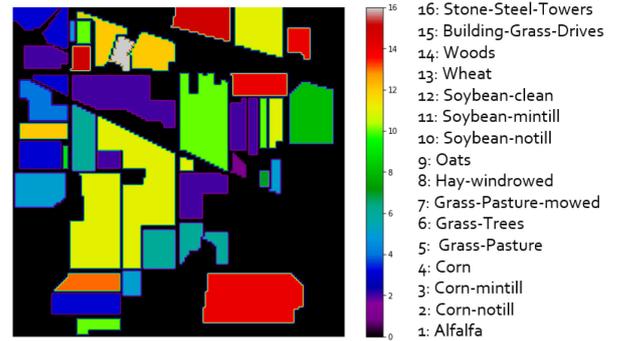


Fig. 1. Indian Pine HSI: Ground truth.

This article is structured as follows. We introduce the basics of hyperspectral imaging in Section II. We present the basics of the information theory and MI-based band subset selection problem in Section III. In Section IV, we discuss the fundamentals of a QUBO problem and demonstrate the *problem mapping* of an MI-based problem to a QUBO-based problem. We introduce the basics of a D-Wave QA and optimize the QUBO-based band subset selection problem for the Indian Pine HSI on a D-Wave QA (see Section V, *quantum optimization*). Finally, we apply the binary quantum classifiers and the novel multilabel classifier to the two- and multilabel dataset in Sections VI, and VII, respectively. Section VIII concludes this article.

II. INTRODUCTION TO HYPERSPECTRAL IMAGING

A hyperspectral imaging sensor mounted on a satellite or aircraft measures the electromagnetic spectrum ranging from the visible to the near infrared wavelengths; for instance, the imaging spectroscopy and the airborne visible/infrared imaging spectrometer (AVIRIS) sensor measures 224 continuous spectral bands ranging from 400 to 2500 nm at 10-nm intervals [17].

As a real-world dataset of HSIs, we consider an Indian Pine HSI obtained by the AVIRIS sensor (see Fig. 1). This low-noise Indian Pine image having the spectral bands of $X = \{\text{band1}, \dots, \text{band200}\}$ elements is a high-dimensional dataset. However, not all of these spectral bands are informative for characterizing a specific class; in other words, some bands of X are redundant or noisy.

It is advantageous to select a highly informative band subset of the given spectral bands for a given class. Hence, we employ an MI-based band subset selection problem as a global optimization problem.

III. INFORMATION THEORY AND MI-BASED BAND SUBSET SELECTION

We select a highly informative band subset for each class of the Indian Pine HSI; for instance, we consider the spectral bands $X = \{X_1, \dots, X_{200}\} = \{\text{band1}, \dots, \text{band200}\}$ of a given class y_S and find its most informative band subset. To find the highly informative band subset for that specific class, we employ an information theory; information is a function of probabilities. Hence, we represent the band X_i and its corresponding class y_S as probabilities. We derived the probabilities

for the band X_i and its class y_S by dividing them into ten bins in a histogram. The probability is then defined as

$$P(X_{n'}) = \frac{X_{n'}}{\sum_{n'=1}^{10} X_{n'}}, \quad P(y_{m'}) = \frac{y_{m'}}{\sum_{m'=1}^{10} y_{m'}} \quad (1)$$

where $X_{n'}$ and $y_{m'}$ represent the number of occurrences of the band X_i and its class y_S in the n' th or m' th bin, respectively. Their joint probability is defined in the same way.

For the selection of the band subset, we exploit *mutual information (MI)* that measures independence between band X_i and its class y_S . It is defined by

$$I(X_i; y_S) = \sum_{m'} \sum_{n'} P(X_{n'}, y_{m'}) \log \frac{P(X_{n'}, y_{m'})}{P(X_{n'})P(y_{m'})} \quad (2)$$

and by *conditional mutual information (CMI)*, which is a measure of the dependence between the band X_i and its class y_S given another band X_j . The CMI can then be written as

$$I(X_i; y_S | X_j) = E(X_i | X_j) - E(X_i | y_S, X_j) \quad (3)$$

where E is the entropy that is a measure of the uncertainty of a random variable [18].

These band subset selection techniques expressed by both (2) and (3) are named after an *MI-based band subset selection* problem, which became popular in machine learning due to its strong mathematical foundation rooted in the information theory.

In the next sections, we pose the MI-based band subset selection problem as a global optimization problem. First, we map our MI-based band subset selection problem to a QUBO problem, and the QUBO problem to a QUBO-based band subset selection problem. Finally, we optimize the QUBO-based band subset selection problem on a D-Wave QA.

IV. PROBLEM MAPPING: THE QUBO-BASED BAND SUBSET SELECTION

A. Mapping of a MI-Based Problem to a QUBO Problem

In this part, we consider and pose the MI-based band subset selection problem as a global optimization problem [11], [12]. Moreover, the maximization over the subsets $\{X_i\}$ can be written as

$$\max_{\{X_i\}} \left[\sum_{X_i} I(X_i; y_S) + \sum_{X_i, X_j} I(X_i; y_S | X_j) \right] \quad (4)$$

where X_i represents the bands of a given class y_S of the Indian Pine HSI (see Fig. 1).

Let us consider the band data $X = \{\text{band1}, \dots, \text{band200}\}$ of a given class of *Alfalfa* or simply $y_S = 1$ as an example case. We assume that (4) is maximized when we use the subset $X_S = \{X_1, X_2\} = \{\text{band1}, \text{band2}\}$. We can express this result in a matrix form such that

$$\begin{aligned} & I(X_1; y_S) + I(X_1; y_S | X_2) + I(X_2; y_S) + I(X_2; y_S | X_1) \Leftrightarrow \\ & \Leftrightarrow \begin{pmatrix} \tilde{x}_1 & \tilde{x}_2 \end{pmatrix} \begin{pmatrix} I(X_1; y_S) & I(X_1; y_S | X_2) \\ I(X_2; y_S | X_1) & I(X_2; y_S) \end{pmatrix} \begin{pmatrix} \tilde{x}_1 \\ \tilde{x}_2 \end{pmatrix} \quad (5) \end{aligned}$$

here, $\tilde{x}_1 = 1$, $\tilde{x}_2 = 1$, and $\tilde{x}_3 = \dots = \tilde{x}_n = 0$. On the other hand, we can interpret this matrix form that the \tilde{x}_n 's are for selecting a highly informative band subset. Hence, we can express the MI-maximization problem expressed by (4) alternatively as

$$\max_{\vec{x}} [\vec{x}^T Q \vec{x}], \quad \vec{x} = (\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)^T, \quad \vec{x} \in \{0, +1\}^n \quad (6)$$

where T represents a transpose operation, and Q is represented diagonal $Q_{ii} = I(X_i; y_S)$ and off-diagonal $Q_{ij} = I(X_i; y_S | X_j)$ elements. We can even transform this maximization problem to a minimization problem by multiplying it by “-1.” As a result, we have

$$\min_{\vec{x}} [\vec{x}^T Q \vec{x}], \quad \vec{x} \in \{0, +1\}^n \quad (7)$$

where $Q_{ii} = -I(X_i; y_S)$ and $Q_{ij} = -I(X_i; y_S | X_j)$ [11]. This form of the minimization problem over binary variables \vec{x} is called a QUBO problem.

The MI-based band subset selection problem is, therefore, equivalent to a QUBO problem when we write “ $-I(X_i; y_S)$ ” and “ $-I(X_i; y_S | X_j)$ ” in the Q matrix, and minimize the Q matrix over the binary variables.

B. Mapping the QUBO Problem to the QUBO-Based Subset Band Selection Problem

To select a highly informative band subset characterizing each class of the Indian Pine image (see Fig. 1), we employ the QUBO problem described by (7) with an additional constraint

$$\min_{\vec{x}} [\vec{x}^T Q \vec{x}], \quad s.t. \quad \sum_{i=1}^n \tilde{x}_i = k, \quad \tilde{x}_i \in \{0, +1\} \quad (8)$$

where k is the number of bands (band subset) of interest, and $n = 200$ is the total number of given bands. Hence, we define the QUBO-based band subset selection problem as

$$\min_{\vec{x}} \left[\vec{x}^T Q \vec{x} + \gamma \sum_{i=1}^n (\tilde{x}_i - k)^2 \right], \quad \vec{x} \in \{0, +1\}^n \quad (9)$$

where γ is a Lagrange multiplier. As an experiment for selecting the most informative band subset for the specific class of an Indian Pine HSI, we consider the band subsets with three elements ($k = 3$).

V. QUANTUM OPTIMIZATION: USING A D-WAVE QA

A. D-Wave QA

We selected a highly informative band subset characterizing the specific class of the Indian Pine HSI by optimizing the QUBO-based band subset selection problem in the form of (9). We optimized this optimization problem on a D-Wave QA, and we even benchmark the D-Wave QA with respect to its conventional version.

A D-Wave QA is a QA for the special class of optimization problems, in particular, QUBO-like problems. Such a QA is a metaheuristic process evolving slowly enough from its initial energy H_i to its final energy H_f in the form of a QUBO problem. The evolution process is expressed by

$$H(t) = (1 - \lambda(t))H_i(\hat{X}) + \lambda(t)H_f(\hat{Z}) \quad (10)$$

Algorithm 1: Fitting Weak Classifiers.

- 1: **INPUT:** Training bands:
 $(\mathbf{x}, \mathbf{y}) = (x_1, y_1), \dots, (x_S, y_S)$; $\triangleright x_S$ represents the three selected bands for a given class y_S (see Table I).
- 2: $\mathbf{y} \in \{-1, +1\}^S$; $\triangleright S$ is the size of the input dataset, and \mathbf{y} represents the two-label of the Indian Pine HSI (see Table III).
- 3: Initialize the weak classifiers: $c = [c_1, \dots, c_N]$; \triangleright DTCs.
- 4: N ; \triangleright the number of DTCs.
- 5: $w_S = (1, \dots, 1)/S$; \triangleright Assigning the same weight to each data element x_S .
- 6: **for** $i \leftarrow 1, \dots, N$ **do**
- 7: Fit a DTC, $c[i]$, to the (\mathbf{x}, \mathbf{y}) with a weight w_S .
- 8: $\mathbf{y}_p = c[i](\mathbf{x})$, $\mathbf{y}_p \in \{-1, +1\}^S$.
- 9: $err_m = w_S \cdot \mathbb{I}(\mathbf{y}_p \neq \mathbf{y}) / \text{sum}(w_S)$.
- 10: $a_m = 0.5 \cdot \log \frac{1 - err_m}{err_m}$.
- 11: $w_S = w_S \cdot \exp(-a_m \cdot \mathbf{y}_p \cdot \mathbf{y})$; \triangleright boosting the weight of misclassified data.
- 12: $w_S = w_S / \text{sum}(w_S)$.
- 13: **end for**
- 14: $h = [h_1, \dots, h_N]$, $h_n \in \mathbb{R}^S$; \triangleright defining an array to store the weak classifier predictions.
- 15: **for** $i \leftarrow 1, \dots, N$ **do**
- 16: $h[i] = c[i](\mathbf{x})$; \triangleright storing the predicted classes.
- 17: **end for**
- 18: $h = h/N$; \triangleright scaling h to the range of $[-1/N, 1/N]$.
- 19: **return** h .
- 20: **STOP ALGORITHM.**

TABLE I

SELECTION OF THE BEST BAND SUBSET FOR EACH CLASS OF THE INDIAN PINE HSI BY USING THE QUBO-BASED BAND SUBSET SELECTION ON A D-WAVE QA

Class labels Y	Selected Bands X_i		
Alfalfa	band41	band47	band77
Corn-notill	band13	band15	band17
Corn-mintill	band3	band133	band190
Corn	band49	band128	band175
Grass-Pasture	band102	band143	band84
Grass-Trees	band23	band40	band53
Grass-Pasture-mowed	band61	band102	band109
Hay-windrowed	band149	band150	band40
Oats	band76	band85	band172
Soybean-notill	band10	band145	band183
Soybean-mintill	band12	band145	band180
Soybean-clean	band4	band12	band136
Wheat	band37	band82	band177
Wood	band63	band102	band190
Building-Grass-Drives	band18	band70	band109
Stone-Steel-Towers	band79	band84	band104

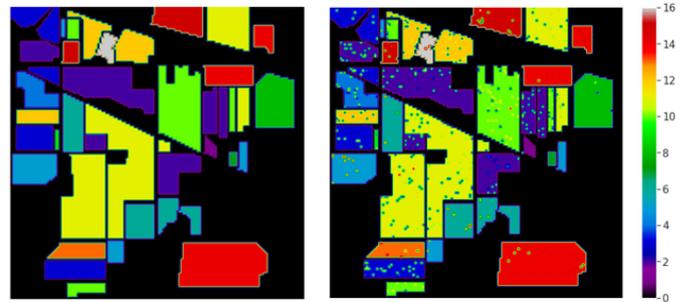


Fig. 2. (Left) Ground truth. (Right) Classification of the $l = 16$ classes characterized by three highly informative spectral bands shown in Table I by using an SVM.

where \hat{X} and \hat{Z} are Pauli- x and $-z$ matrices, H_i is the initial Hamiltonian of a system for a given time function of $\lambda(t) = 0$, and H_f is the QUBO problem with $\lambda(t) = 1$ [1]–[3].

The hardware of the D-Wave QA has a specific graph topology $G = (V, E)$ named *Pegasus*; its vertices represent binary variables \vec{x} , and its edges define interaction strengths among the binary variables. However, the connectivity of these binary variables in the *Pegasus* topology is very constrained; in particular, only the certain binary variables are allowed to interact with others through the edges [19].

In addition, the performance of a D-Wave QA strongly depends on mapping the binary variables of our QUBO problem expressed by (9) to the *Pegasus* topology. As it is possible to map (embed) our QUBO problem to the *Pegasus* topology as efficiently as possible, we employed a technique called *minor embedding*, which is offered by the company D-Wave systems [2], [19].

B. Quantum Optimization for the Band Subset Selection

Quantum optimization is an optimization of our QUBO-based band subset selection problem on a D-Wave QA. We performed our experiment in a classical annealer and a D-Wave QA. Both of these annealers selected the same band subset for each class

of the Indian Pine HSI; we show these selected band subsets in Table I, while $k = 3$ in (9).

To prove that we selected the highly informative band subset for each class on a D-Wave QA, we performed the scene classification for our Indian Pine HSI by using a DTC and an SVM shown in Fig. 2 as a proof-of-concept.

In addition, we discovered that we needed at least a 10-D parameter to reach the same accuracy as our proof-of-concept method when we apply the PCA for the dimensionality reduction and conventional classifiers (the DT and the SVM) for the multiclass classification of the Indian Pine HSI (see Table I) [8]. For this scenario, we present the classification accuracy of the test dataset in Table II.

These findings lead to the conclusion that our QUBO-based band subset selection method identified the highly informative band subset, and it even helped to reduce a storage space and the computational load for training the given classifiers.

VI. CASE STUDY OF A BINARY QUANTUM CLASSIFIER ON A D-WAVE QA FOR HSI

We have the Indian Pine HSI with 16 classes, where each class is characterized by three highly informative bands selected by our QUBO-based band subset selection method shown in

Algorithm 2: Qboost Classifier.

- 1: **INPUT:** h from (Algorithm 1) or given.
 - 2: **OUTPUT:** The strong classifier C .
 - 3: Fit the weak classifiers to (\mathbf{x}, \mathbf{y}) (Algorithm 1): h (if h is not given).
 - 4: **The weak subset classifier selection:**
 $\alpha^* = \min_{\alpha} [\alpha^T Q \alpha]$, $\alpha \in \{0, +1\}^N$; \triangleright QUBO problem
 - 5: **Diagonal and off-diagonal elements of a matrix Q :**
 - 6: $Q \in \mathbf{R}^{N \times N}$.
 - 7: **for** $i \leftarrow 1, \dots, N$ **do**
 - 8: $Q_{ii} = S/N^2 + \lambda - 2(h[i])^T \cdot \mathbf{y}$.
 - 9: **end for**
 - 10: **for** $i \leftarrow 1, \dots, N$ **do**
 - 11: **for** $j \leftarrow i + 1, \dots, N$ **do**
 - 12: $Q_{ij} = (h[i])^T \cdot h[j]$.
 - 13: **end for**
 - 14: **end for**
 - 15: **Optimize QUBO problem on a D-Wave QA**
 - 16: An optimal estimator weight vector: α^* .
 - 17: **PREDICT:** given the test band set (x_1, \dots, x_t) ;
 - 18: $T = \frac{1}{t} \sum_{t=1}^t \sum_{i=1}^N \alpha^* h[i](x_t)$.
 $C(x_t) = \text{sign}(\sum_{i=1}^N \alpha^* [i] h[i](x_t) - T)$.
 - 19: **STOP ALGORITHM.**
-

TABLE II
CLASSIFICATION ACCURACY OF THE DTC AND THE SVM

QUBO-based band selection			PCA	
Classifier	DTC	SVM	DTC	SVM
Accuracy	0.74	0.81	0.72	0.81

Table I. In this section, we analyze binary quantum classifiers, namely a Qboost classifier, and a Qboost-Plus classifier, for a two-label dataset of the Indian Pine HSI created as the binary output of a D-Wave QA; for instance, *Alfalfa and Corn-notill*, or *Corn-mintill and Corn-notill*, etc. Further, we benchmarked the classification accuracy of our binary quantum classifiers with respect to conventional binary classifiers, such as a DTC, an SVM, and an Adaboost classifier.

We considered first two types of boosting algorithms, a Qboost and an Adaboost algorithm (classifier). The Qboost classifier is a quantum version of an Adaboost classifier. Here, we use two types of terminology for these classifiers, a strong classifier C and a weak classifier c_i . The strong classifier leverages many weak classifiers to achieve its high classification accuracy; the weak classifier is a classifier that classifies a given dataset better than random guessing [20].

A. Basics of an Adaboost Classifier

An Adaboost classifier is an algorithm for finding an optimal estimator weight of many weak classifiers so that the classifier C is maximized [21]

$$C(x_S) = \text{sign} \left[\sum_{i=1}^N \alpha_i c_i(x_S) \right], \quad c_i(x_S) \in \{-1, +1\} \quad (11)$$

where (x_S, y_S) represents a training dataset, and $\alpha_i \in [0, +1]$ is the estimator weight that is continuous-valued. Here, $\text{sign}(f(x_S)) = 1$ if $f(x_S) > 0$, $\text{sign}(f(x_S)) = -1$ if $f(x_S) < 0$, and $\text{sign}(f(x_S)) = 0$ otherwise. The loss of the Adaboost classifier is defined as an exponential loss

$$\alpha^* = \min_{\alpha} \left[\sum_{s=1}^S \exp -y_s \sum_{i=1}^N \alpha_i c_i(x_s) / S \right]. \quad (12)$$

In contrast, a Qboost classifier is an algorithm for finding an optimal estimator weight that takes only binary numbers $\alpha_i \in \{0, +1\}$, and its loss is defined by a squared loss denoted as L_2 . Hence, the Qboost classifier is equivalent to a subset selection algorithm among many weak classifiers to approximately maximize the accuracy of the strong classifier. In next section, we delve into the Qboost classifier in more detail.

In general, these boosting algorithms start with assigning identical weights w_S to our dataset. The weak classifiers classify these datasets, and if the data are misclassified, then the weight of that data are increased (boosted). This procedure is repeated until no further improvement in the classification accuracy can be seen. A DTC with a depth of one is considered as a weak classifier; sometimes, it is called a decision stump classifier. We already presented the steps for boosting a weight w_S and the weak classifier in Algorithm 1.

B. Qboost Classifier for a Two-Label Dataset of the Indian Pine HSI

Moving toward the Qboost classifier, after having stopped boosting the weight of our dataset, the Qboost classifier selects the weak subset classifier so that the classification accuracy of the strong classifier is maximized. We executed the weak subset classifier selection algorithm on a D-Wave QA as shown in Algorithm 2. Below, we explain the derivation of Algorithm 2 in detail. More importantly, the Qboost classifier exploits the weight boosting by solving the weak subset classifier selection problem on a D-Wave QA.

For the two-label dataset of the Indian Pine HSI, we define the training band dataset as $(x_1, y_1), \dots, (x_S, y_S)$, the test band dataset as (x_1, \dots, x_t) , and the strong classifier, $C(x_S) \in \{-1, +1\}$, which is a binary classifier in the form of [13] and [20]

$$C(x_S) = \text{sign} \left[\sum_{i=1}^N \alpha_i c_i(x_S) \right], \quad c_i(x_S) \in \{-1, +1\} \quad (13)$$

where $\alpha_i \in \{0, 1\}$ is the estimator weight, and $c_i(x_S)$ is the weak classifier; we chose DTCs as our weak classifiers.

Recent papers on theoretical studies [13], [14] and a practical application for the remote sensing [22] are proposed to formulate the loss of the strong classifier as a squared loss L_2

$$\alpha^* = \min_{\alpha_i, \lambda} \left[\sum_{s=1}^S \left(\sum_{i=1}^N \alpha_i c_i(x_s) - y(x_s) \right)^2 + \lambda \sum_{i=1}^N \alpha_i^0 \right] \quad (14)$$

where α^* represents the optimal estimator weight vector, S is the size of the training band dataset, and $\lambda \sum_{i=1}^N \alpha_i^0$ represents a

0-norm term. By expanding the squared loss function, we have

$$\alpha^* = \min_{\alpha, \lambda} \left[\sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j \left(\sum_{s=1}^S c_i(x_s) c_j(x_s) \right) + \sum_{i=1}^N \alpha_i \left(\lambda - 2 \sum_{s=1}^S c_i(x_s) y(x_s) \right) \right] \quad (15)$$

which is in the form of a QUBO problem, while we define

$$Q_{ij} = \sum_{s=1}^S c_i(x_s) c_j(x_s) \quad (16)$$

$$Q_{ii} = S/N^2 + \lambda - 2 \sum_{s=1}^S c_i(x_s) y(x_s).$$

Then, we can write

$$\alpha^* = \min_{\alpha} [\alpha^T Q \alpha], \quad \alpha \in \{0, +1\}^N. \quad (17)$$

We optimized this problem on a D-Wave QA to select the weak subset classifier in its quadratic form.

Then, we obtained the optimal estimator weight vector α^* , and the strong classifier for the test band dataset becomes

$$C(x_t) = \text{sign} \left[\sum_{i=1}^N \alpha_i^* c_i(x_t) - T \right]$$

$$T = \frac{1}{t} \sum_{t=1}^t \sum_{i=1}^N \alpha_i^* c_i(x_t) \quad (18)$$

where (x_1, \dots, x_t) are from the test band dataset of the Indian Pine HSI, and T is derived experimentally to increase the classification accuracy of the strong classifier $C(x_t)$ [13], [14]. We have already presented the procedures of the Qboost classifier in Algorithms 1 and 2.

Second, we chose the *DTC*, *SVM*, and *Qboost* classifiers as weak classifiers instead of only a *DTC*. This method is sometimes called an ensemble method. By exploiting (13) and (15), we again formulated weak classifiers such that

$$C(x_S) = \text{sign} \left[\sum_{i=1}^3 \alpha_i c_i(x_S) \right], \quad c_i(x_S) \in \{-1, +1\} \quad (19)$$

where $c_1(x_S)$, $c_2(x_S)$, and $c_3(x_S)$ represent the *DTC*, *SVM*, and *Qboost* classifiers, respectively. In this scenario, we have $h = [c_1(x_S), c_2(x_S), c_3(x_S)]$ in Algorithm 2, and this ensemble method is called a *Qboost-Plus* classifier [2].

C. Benchmarking *Qboost* and *Qboost-Plus* for the Two-Class Classification

We run our experiment in several scenarios for the two-label dataset of the Indian Pine HSI by using the *DTC*, *SVM*, *Qboost*, *Qboost-Plus*, and *Adaboost* classifier. These scenarios are as follows.

- 1) *DTC* for the two-label dataset of the Indian HSI.
- 2) *SVM* for the two-label dataset of the Indian HSI.

TABLE III

CLASSIFICATION ACCURACY OF THE *DTC*, *SVM*, *Qboost*, *Qboost-Plus*, AND *ADABOOST* FOR THE TWO-LABEL OF THE INDIAN PINE HSI; $\{i, j\}$ REPRESENTS THE TWO-LABELS, W.G., $\{1, 2\} \rightarrow$ ALFALFA AND CORN-NOTILL (SEE FIG. 1)

Binary Classifier Accuracy					
Classes	DTC	SVM	Qboost	Qboost-Plus	Adaboost
{1, 2}	0.99	0.99	0.99	0.99	0.99
{2, 3}	0.89	0.83	0.64	0.85	0.84
{3, 4}	0.88	0.92	0.83	0.92	0.90
{4, 5}	0.95	0.98	0.95	0.98	0.96
{5, 6}	0.99	0.99	0.98	0.99	0.98
{6, 7}	1.00	1.00	1.00	1.00	1.00
{7, 8}	0.95	0.99	0.99	0.99	0.99
{8, 9}	1.00	1.00	1.00	1.00	1.00
{9, 10}	1.00	0.99	0.99	0.99	0.99
{10, 11}	0.75	0.78	0.72	0.75	0.78
{11, 12}	0.83	0.86	0.83	0.86	0.84
{12, 13}	1.00	1.00	1.00	1.00	1.00
{13, 14}	0.99	0.99	0.99	0.99	0.99
{14, 15}	0.85	0.90	0.87	0.90	0.88
{15, 16}	0.99	0.99	0.99	0.99	0.99

By a **Bold** font, we noted the highest accuracy value of the Qboost-Plus classifier with respect to the Adaboost classifier.

- 3) *Qboost* with 30 weak classifiers for the two-label dataset of the Indian HSI; the weak classifiers are the DT classifiers with the depth three.
- 4) *Qboost-Plus* for the two-label dataset of the Indian HSI; the weak classifiers are a *DTC*, an *SVM*, and a *Qboost* classifier.
- 5) *Adaboost* with 30 weak classifiers for the two-label dataset of the Indian HSI; the weak classifiers are the decision stump classifiers.

All aforementioned scenarios used for benchmarking are the two-label classification of the Indian Pine HSI, and we present the classification accuracy of our experiment in Table III. We even compared the boosting algorithms, the *Qboost-Plus* and the *Adaboost* classifier. Their results demonstrate that the *Qboost-Plus* classifier performs the same as the *Adaboost* classifier and even better in some instances.

In this part, we selected the most highly informative band of the Indian Pine HSI by using our QUBO-based band subset selection method. Furthermore, we leveraged these selected bands to benchmark our *Qboost* and *Qboost-plus* algorithms with respect to the classical classifiers. Our quantum classifiers clearly outperform the conventional classifiers for most of the binary instances of the Indian pine HSI.

VII. NOVEL MULTILABEL CLASSIFIER FOR THE INDIAN PINE HSI ON A D-WAVE QA

In the prior section, we exhibited that our quantum binary classifiers (*Qboost* and *QboostPlus*) classify the two-label of the Indian Pine HSI with high accuracy due to the binary output of a D-Wave QA. However, the Indian Pine HSI has 16 classes, and the quantum binary classifiers are needed to extend for the multilabel classification. Hence, we propose a novel technique for the multilabel classification via an ECOC, and namely, we leverage an ECOC technique to classify the multilabel of the Indian Pine HSI when using our binary quantum classifiers [15], [16], [23].

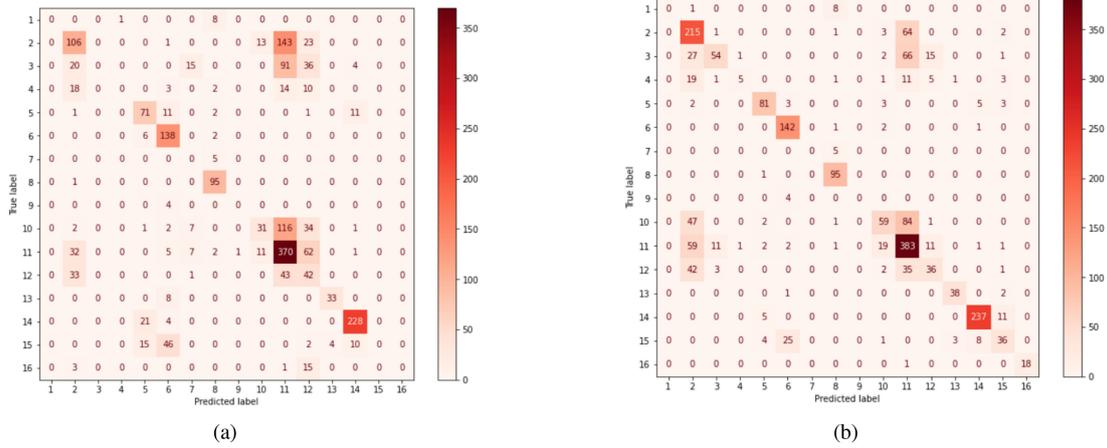


Fig. 3. Confusion matrix for the $l = 16$ labels of the Indian Pine HSI. (a) *Qboost-Plus* classifier via the ECOC technique. (b) *Adaboost* classifier via the ECOC technique.

Algorithm 3: A Multilabel Classifier by Using Quantum Binary Classifiers Via an ECOC.

- 1: **INPUT:** Training bands:
 $(\mathbf{x}, \mathbf{y}) = (x_1, y_1), \dots, (x_S, y_S)$; $\triangleright x_S$ represents the three selected bands for a given class y_S (see Table I).
 $\mathbf{y} \in \{1, 2, \dots, 16\}^S$;
 $\triangleright \mathbf{y}$ represents $l = 16$ distinct labels of the Indian Pine HSI, and S is a size of the training dataset.
 - 2: **OUTPUT:** Quantum binary classifiers:
 $C_b = \{C_1, C_2, \dots, C_{24}\}$.
 - 3: **CODING MATRIX:**
 - 4: Assign $b = 24$ codewords to each class ($b > \log_2 l$), and generate l by b coding matrix M for $l = 16$ distinct labels (see Table IV).
 - 5: Construct S by b coding matrix M' for training classes \mathbf{y} .
 - 6: **TRAINING:**
 - 7: **for** $i \leftarrow 1, \dots, b$ **do**
 - 8: Construct two sets, G_i and \overline{G}_i . G_i consists of all labels for which $M'[:, i] = 1$, and \overline{G}_i is the complement set.
 - 9: Fit a quantum binary classifier C_i to distinguish G_i from \overline{G}_i by using Algorithm 2.
 - 10: **end for**
 - 11: **TESTING:**
 - 12: Given an unlabeled data x_t .
 - 13: Evaluate the trained quantum binary classifiers
 $C_b(x_t) = \{C_1(x_t), C_2(x_t), \dots, C_{24}(x_t)\}$ by employing the step 17 of Algorithm 2.
 - 14: Compute an Euclidean/Hamming distance:
 $d_j = d(C_b(x_t), M[j, :])$, $j = 1, 2, \dots, l$.
 - 15: **return** $\text{argmin}_j d_j$; \triangleright codewords (a label) for the unlabeled data x_t .
 - 16: **STOP ALGORITHM.**
-

TABLE IV

EXAMPLE OF THE CODING MATRIX M FOR THE ALL $l = 16$ LABELS OF THE INDIAN PINE HSI GENERATED RANDOMLY, AND EACH CLASS IS CHARACTERIZED BY $b = 24$ CODEWORDS

Classes	A coding matrix M with $l \times b$ elements.												
1	1	1	0	1	0	1	0	0	0	0	1	...	1
2	0	1	1	0	1	0	0	0	1	1	1	...	0
...
16	0	1	0	0	1	1	1	1	1	0	1	...	1

A. *ECOC Technique* (see Algorithm 3 for a Detailed Procedure).

- 1) *Coding matrix:* We assign unique b -bits (codewords) to each class of the Indian Pine HSI such that $b > \log_2 l$ where $l = 16$ is a number of classes; the classes are represented by a so-called coding matrix $M \in \{0, 1\}^{l \times b}$ (see Table IV), and $M' \in \{0, 1\}^{S \times b}$ for a training dataset with size S . In our case, each class is represented by $b = 24$ codewords generated randomly.
- 2) *Training:* We train each column of the coding matrix M' by quantum binary classifiers $C_b = \{C_1, C_2, \dots, C_{24}\}$.
- 3) *Testing:* For an unlabeled input x_t , we evaluate $C_b(x_t) = \{C_1(x_t), C_2(x_t), \dots, C_{24}(x_t)\}$, and then, we assign $C_b(x_t)$ to the closest codewords in the coding matrix M by using an Euclidean/Hamming distance.

B. *Benchmarking Qboost and Qboost-Plus for the Multilabel Classification*

We run our experiment for the multilabel of the Indian Pine HSI via the ECOC by using the *DTC*, *SVM*, *Qboost*, *Qboost-Plus*, and *Adaboost* classifier. Furthermore, we presented the classification accuracy of our experiment in Table V. We compared also the classification accuracy and the confusion matrix of the *Qboost-Plus* with one of the *Adaboost* classifier (see Fig. 3). Their results again demonstrate that the *Qboost-Plus* classifier

TABLE V
CLASSIFICATION ACCURACY OF THE DTC, SVM, QBOOST CLASSIFIER,
QBOOST-PLUS CLASSIFIER, AND ADABOOST CLASSIFIER FOR THE ALL 16
LABELS OF THE INDIAN PINE HSI; {1, 2, . . . , 16} REPRESENTS THE ALL 16
LABELS (SEE FIG. 1)

A multi-label classifier accuracy					
Classes	DTC	SVM	Qboost	Qboost-Plus	Adaboost
{1, 2, . . . , 16}	0.82	0.72	0.67	0.77	0.64

By a **Bold** font, we noted the highest accuracy value of the Qboost-Plus classifier with respect to the Adaboost classifier.

beats the *Adaboost* classifier when we leverage the ECOC technique for a multilabel classification case. More importantly, we provided a novel multilabel classifier via the ECOC technique when applying a quantum computing device yielding binary outputs.

VIII. DISCUSSION AND CONCLUSION

In the first part of this article, we used an MI-based band subset selection technique as a global optimization approach for a real-world problem of the Indian Pine hyperspectral dataset on a D-Wave QA. We first mapped this MI-based band subset selection problem to a QUBO-based band subset selection problem. Then, we benchmarked and assessed the performance of a D-Wave QA compared to a conventional annealer. We demonstrated that the D-Wave QA correctly selects highly informative bands competitive to a conventional annealer. To prove that our D-Wave QA selected the best bands for each class, we classified all 16 classes based on their three highly informative bands by applying a DTC and a SVM classifier. Their classification results exhibit that the selected bands are the highly informative ones. Besides, the feature selection method saves storage space and reduces the computational load for the training process.

In the second part of our article, we first tested a binary classification for the Indian Pine HSI due to the binary output of our D-Wave QA. We proposed to employ two binary quantum classifiers, Qboost and Qboost-Plus, to our two-label dataset. Second, we provided an ECOC for the multilabel classification of the Indian Pine HSI when applying our binary quantum classifiers. Here, the classes are characterized by the bands selected during the first part of our study. We benchmarked these binary quantum classifiers and the novel multilabel classifier in comparison to conventional classifiers that are a DTC, a SVM classifier, and an Adaboost classifier. Our binary quantum classifiers and our novel multilabel classifier even outperform these conventional classifiers for most instances of the two- and multilabel dataset.

In the end, we realized how to leverage a quantum annealing device to extract knowledge and support real-world optimization problems in comparison to conventional machine learning techniques. In addition, we conceived strategies for formulating and embedding real-world problems to the topology of a D-Wave machine.

We must note, however, that our method is not intended to compete with a conventional method, but we intended to find a proper dataset in Earth observation to evaluate an existing

quantum algorithm on a D-Wave QA or the future quantum computers since the choice and the size of a dataset play a vital role in quantum computers.

In terms of a future work, we will design a hybrid quantum-classical network for Earth observation datasets, which exploits both quantum computers (a QA and gate-based quantum computer) and a conventional computer. Such a hybrid network will be independent of the choice and size of datasets.

ACKNOWLEDGMENT

The authors would like to thank G. Schwarz (DLR, Oberpfaffenhofen) for his valuable comments and contributions for enhancing a quality of this article.

REFERENCES

- [1] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, "Quantum computation by adiabatic evolution," *arXiv:0001106*, Jan. 2000. [Online]. Available: <https://arxiv.org/abs/quant-ph/0001106>
- [2] *A D-Wave Quantum Annealer*, 2021. [Online]. Available: <https://cloud.dwavesys.com/leap>
- [3] V. S. Denchev *et al.*, "What is the computational value of finite-range tunneling?" *Phys. Rev. X*, vol. 6, Aug. 2016, Art. no. 031015. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevX.6.031015>
- [4] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, "A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem," *Science*, vol. 292, no. 5516, pp. 472–475, 2001. [Online]. Available: <https://science.sciencemag.org/content/292/5516/472>
- [5] T. Stollenwerk *et al.*, "Quantum annealing applied to de-conflicting optimal trajectories for air traffic management," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 1, pp. 285–297, Jan. 2020.
- [6] E. Rieffel, D. Venturelli, B. O’Gorman, M. Do, E. Prystay, and V. Smelyanskiy, "A case study in programming a quantum annealer for hard operational planning problems," *Quantum Inf. Process.*, vol. 14, pp. 1–36, 2015.
- [7] F. Melgani and L. Bruzzone, "Classification of hyperspectral remote sensing images with support vector machines," *IEEE Trans. Geosci. Remote Sens.*, vol. 42, no. 8, pp. 1778–1790, Aug. 2004.
- [8] S. Prasad and L. M. Bruce, "Limitations of principal components analysis for hyperspectral target recognition," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 4, pp. 625–629, Oct. 2008.
- [9] B. Guo, S. R. Gunn, R. I. Damper, and J. D. B. Nelson, "Band selection for hyperspectral image classification using mutual information," *IEEE Geosci. Remote Sens. Lett.*, vol. 3, no. 4, pp. 522–526, Oct. 2006.
- [10] S. Sarmah and S. K. Kalita, "Mutual information-based hierarchical band selection approach for hyperspectral images," in *Advances in Electronics, Communication and Computing*, A. Kalam, S. Das, and K. Sharma, Eds. Singapore: Springer, 2018, pp. 755–763.
- [11] X. V. Nguyen, J. Chan, S. Romano, and J. Bailey, "Effective global approaches for mutual information based feature selection," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2014, pp. 512–521. [Online]. Available: <https://doi.org/10.1145/2623330.2623611>
- [12] N. X. Vinh, S. Zhou, J. Chan, and J. Bailey, "Can high-order dependencies improve mutual information based feature selection?" *Pattern Recognit.*, vol. 53, pp. 46–58, 2016. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0031320315004276>
- [13] H. Neven, V. S. Denchev, G. Rose, and W. G. Macready, "Qboost: Large scale classifier training with adiabatic quantum optimization," in *Proc. Asian Conf. Mach. Learn.*, Nov. 2012, vol. 25, pp. 333–348. [Online]. Available: <http://proceedings.mlr.press/v25/neven12.html>
- [14] V. S. Denchev, N. Ding, S. V. N. Vishwanathan, and H. Neven, "Robust classification with adiabatic quantum optimization," *arXiv:1205.1148*, May 2012. [Online]. Available: <https://arxiv.org/abs/1205.1148>
- [15] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *J. Artif. Int. Res.*, vol. 2, no. 1, pp. 263–286, Jan. 1995.

- [16] A. Radoi and M. Datcu, "Multilabel annotation of multispectral remote sensing images using error-correcting output codes and most ambiguous examples," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 7, pp. 2121–2134, Jul. 2019.
- [17] *Airborne visible/infrared Imaging Spectrometer (AVIRIS)*, 2021. [Online]. Available: <https://aviris.jpl.nasa.gov/>
- [18] S. Barnett, Ed., *Quantum Information*. Oxford, U.K.: Oxford Univ. Press, 2009.
- [19] K. Boothby, P. Bunyk, J. Raymond, and A. Roy, "Next-generation topology of D-wave quantum processors," *arXiv:2003.00133*, Feb. 2020. [Online]. Available: <https://arxiv.org/abs/2003.00133>
- [20] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S002200009791504X>
- [21] T. Hastie, S. Rosset, J. Zhu, and H. Zou, "Multi-class adaboost," *Statist. Interface*, vol. 2, no. 3, pp. 349–360, 2009.
- [22] E. Boyda, S. Basu, S. Ganguly, A. Michaelis, S. Mukhopadhyay, and R. R. Nemani, "Deploying a quantum annealing processor to detect tree cover in aerial imagery of California," *PLOS One*, vol. 12, no. 2, pp. 1–22, Feb. 2017. [Online]. Available: <https://doi.org/10.1371/journal.pone.0172505>
- [23] A. Berger, "Error-correcting output coding for text classification," in *Proc. IJCAI-99 Workshop Mach. Learn. Inf. Filtering*, Jul. 2001.

Soronzonbold Otgonbaatar received the B.Sc. degree from the National University of Mongolia, Ulaanbaator, Mongolia, and the M.Sc. degree in theoretical physics from the University of Siegen, Siegen, Germany, in 2014 and 2016, respectively. He is currently working toward the Ph.D. degree with German Aerospace Center, Wessling, Germany.

His research interests include computational science, artificial intelligence (i.e., machine learning), quantum computing and algorithms, and quantum machine learning with practical applications in remote sensing data.

Mihai Datcu (Fellow, IEEE) received the M.S. and Ph.D. degrees in electronics and telecommunications from the University Politehnica of Bucharest (UPB), Bucharest, Romania, in 1978 and 1986, respectively, and the habilitation a Diriger Des Recherches degree in computer science from the University Louis Pasteur, Strasbourg, France, in 1999.

Since 1981, he has been a Professor with the Department of Applied Electronics and Information Engineering, Faculty of Electronics, Telecommunications and Information Technology, UPB. Since 1993, he has been a Scientist with the German Aerospace Center (DLR), Wessling, Germany. He has held Visiting Professor appointments with the University of Oviedo, Spain, the University Louis Pasteur and the International Space University, both in Strasbourg, France, the University of Siegen, Germany, the University of Innsbruck, Austria, the University of Alcala, Spain, the University Tor Vergata, Rome, Italy, the University of Trento, Italy, the Unicamp, Campinas, Brazil, the China Academy of Science, Shenyang, China, the Universidad Pontificia de Salamanca, Campus de Madrid, Spain, the University of Camerino, Italy, and the Swiss Center for Scientific Computing, Manno, Switzerland. From 1992 to 2002, he had an Invited Professor Assignment with the Swiss Federal Institute of Technology, Switzerland. Since 2001, he had been initiating and leading the Competence Center on Information Extraction and Image Understanding for Earth Observation, Paris Institute of Technology, ParisTech, France, a collaboration of DLR with the French Space Agency. He has been a Professor holder of the DLR-CNES Chair with ParisTech. He has initiated the European frame of projects for image information mining and is involved in research programs for information extraction, data mining and knowledge discovery, and data science with the ESA, NASA, and in a variety of national and European projects. He is the Director of the Research Center for Spatial Information, UPB. He is a Senior Scientist and the Data Intelligence and Knowledge Discovery Research Group Leader with the Remote Sensing Technology Institute, DLR, and a delegate with the DLR-ONERA Joint Virtual Center for AI in Aerospace. He is a member of the ESA Working Group Big Data from Space and Visiting Professor with ESA's Phi-Lab. His research interests include explainable and physics-aware artificial intelligence, smart radar sensors design, and quantum machine learning with applications in Earth observation.

Dr. Datcu was the recipient of the National Order of Merit with the rank of Knight, for outstanding international research results, awarded by the President of Romania, in 2008, and the Romanian Academy Prize Traian Vuia for the development of the SAADI image analysis system and his activity in image processing in 1987. He was awarded the Chaire D'excellence Internationale Blaise Pascal 2017 for data science in Earth observation and the 2018 Ad Astra Award for Excellence in Science. He has served as a Coorganizer for international conferences and workshops and as a Guest Editor for the IEEE and other journals. He is representative of Romanian in the Earth Observation Program Board.

4.2.3 Publication F: Assembly of a Coreset of Earth Observation Images on a Small Quantum Computer

S. Otgonbaatar and M. Datcu

Abstract. Until now, we have designed and conducted our experiments for large-scale classical datasets and small quantum computers, including [NISQ](#) and [QA](#) devices, as presented in **Publication A** to **Publication E**. At the time of writing, [NISQ](#) computers have around 100 noisy qubits, and the D-Wave device has approximately 5,000 qubits arranged in a particular restricted topology. **Publication A** to **Publication E** proposed and invented novel quantum feature selection and quantum classifiers using big satellite datasets. We even benchmarked the performance of the small quantum computer compared with conventional classical computer. However, the challenge of computing optimization problems involving large-scale datasets on a small quantum computer still exists. Hence, this **Publication F** introduces the concept of a “coreset”. The coreset is a small, representative weighted subset of the original dataset. It opens up an opportunity for training quantum classifiers on large-scale datasets employing a quantum computer with a limited number of input qubits and depth. Our method involves optimizing a weighted [SVM](#) on the coreset of a big dataset while employing the D-Wave quantum annealer. We select the coresets of two-label classes of the Indian Pine hyperspectral image and the [PolSAR](#) image of California. We then train both a [QSVM](#) and a [Classical Support Vector Machine \(CSVM\)](#) on the coresets on the D-Wave system according to the workflow presented in [Chapter 3.2.1](#) of this dissertation. Our experiments show that we correctly assemble the coresets of our datasets validated by KL-divergence values (the smaller, the better). Moreover, we benchmark the performance of the [QSVM](#) compared with the [CSVM](#). The results show that the quantum classifier trained on coresets yields competitive classification values with respect to their classical counterparts. In particular, we propose and experiment with a novel machine learning method for training quantum classifiers on large-scale datasets via a coreset on a small quantum computer. We also gain insights into how to train [AI](#) models on big datasets on a future quantum computer.

Article

Assembly of a Coreset of Earth Observation Images on a Small Quantum Computer

Soronzonbold Otgonbaatar ^{1,†,*}  and Mihai Datcu ^{1,2,†}¹ German Aerospace Center (DLR), 82234 Oberpfaffenhofen, Germany; Mihai.Datcu@dlr.de² Department of Computers, Politehnica University of Bucharest (UPB), 060042 Bucharest, Romania

* Correspondence: soronzonbold.otgonbaatar@dlr.de

† Current address: DLR, Münchener Straße 20, 82234 Weßling, Germany

Abstract: Satellite instruments monitor the Earth's surface day and night, and, as a result, the size of Earth observation (EO) data is dramatically increasing. Machine Learning (ML) techniques are employed routinely to analyze and process these big EO data, and one well-known ML technique is a Support Vector Machine (SVM). An SVM poses a quadratic programming problem, and quantum computers including quantum annealers (QA) as well as gate-based quantum computers promise to solve an SVM more efficiently than a conventional computer; training the SVM by employing a quantum computer/conventional computer represents a quantum SVM (qSVM)/classical SVM (cSVM) application. However, quantum computers cannot tackle many practical EO problems by using a qSVM due to their very low number of input qubits. Hence, we assembled a coreset ("core of a dataset") of given EO data for training a *weighted* SVM on a small quantum computer, a D-Wave quantum annealer with around 5000 input quantum bits. The coreset is a small, representative weighted subset of an original dataset, and its performance can be analyzed by using the proposed *weighted* SVM on a small quantum computer in contrast to the original dataset. As practical data, we use *synthetic* data, *Iris* data, a *Hyperspectral Image* (HSI) of Indian Pine, and a *Polarimetric Synthetic Aperture Radar* (PolSAR) image of San Francisco. We measured the closeness between an original dataset and its coreset by employing a Kullback–Leibler (KL) divergence test, and, in addition, we trained a *weighted* SVM on our coreset data by using both a D-Wave quantum annealer (D-Wave QA) and a conventional computer. Our findings show that the coreset approximates the original dataset with very small KL divergence (smaller is better), and the *weighted* qSVM even outperforms the *weighted* cSVM on the coresets for a few instances of our experiments. As a side result (or a by-product result), we also present our KL divergence findings for demonstrating the closeness between our original data (i.e., our synthetic data, *Iris* data, hyperspectral image, and PolSAR image) and the assembled coreset.

Keywords: coreset assembly; quantum support vector machines; hyperspectral images; PolSAR images; quantum machine learning



check for updates

Citation: Otgonbaatar, S.; Datcu, M. Assembly of a Coreset of Earth Observation Images on a Small Quantum Computer. *Electronics* **2021**, *10*, 2482. <https://doi.org/10.3390/electronics10202482>

Academic Editor: Jian-Qiang You

Received: 30 August 2021

Accepted: 11 October 2021

Published: 12 October 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 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

Remotely sensed images are used for EO and acquired by means of aircraft or satellite platforms. The acquired images from satellites are available in digital format and consist of information on the number of spectral bands, radiometric resolution, spatial resolution, etc. A typical EO dataset is big, massive, and hard to classify by using ML techniques when compared with conventional non-satellite images [1,2]. In principle, ML techniques are a set of methods for recognizing and classifying common patterns in a labeled/unlabeled dataset [3,4]. However, they are computationally expensive and intractable to train big massive data. Recently, several studies proposed to use only a coreset ("core of a dataset") of an original dataset for training ML methods and tackling intractable posterior distributions via Bayesian inference [5–7], even for a Support Vector Machine (in short, SVM) [8]. The coreset is a small, representative weighted subset

of an original dataset, and ML methods trained on the coresets yield results being competitive with the ones trained on the original dataset. The concept of a coreset opens a new paradigm for training ML models by using small quantum computers [9,10] since currently available quantum computers offered by *D-Wave Systems* (D-Wave QA) and by *IBM quantum experience* (a gate-based quantum computer) comprise very few quantum bits (qubits) (<https://cloud.dwavesys.com/leap>, <https://quantum-computing.ibm.com/>, accessed on 30 August 2021). In particular, quantum computers promise to solve some intractable problems in ML [11–13], and to train an SVM even better/faster than a conventional computer when its input data volume is very small (“core of a dataset”) [14,15]. Training ML methods by using a quantum computer or by exploiting quantum information is called Quantum Machine Learning (QML) [16–18], and finding the solutions of the SVM on a quantum computer is termed a quantum SVM (qSVM), otherwise classical SVM (cSVM).

This work uses a D-Wave QA for training a weighted SVM since the D-Wave QA promises to solve a quadratic programming problem, and our method can be easily adapted and extended for a gate-based quantum computer. The D-Wave QA has a very small number of input qubits (around 5000) and a specific *Pegasus* topology for the connectivity of its qubits [19], and it is solely designed for solving a Quadratic Unconstrained Binary Optimization (QUBO) problem [12,20]. For practical EO data, there is a benchmark and a demonstration example for training an SVM with binary quantum classifiers when using a D-Wave QA [21,22]. Here, the SVM is a quadratic programming problem considered as a QUBO problem. Furthermore, there is a challenge to embed the variables of a given SVM problem into the *Pegasus* topology (i.e., the connectivity constraint of qubits), and to overcome this constraint of a D-Wave QA, the authors of [21] employed a *k-fold* approach to their EO data such that the size of variables in the SVM satisfies the connectivity constraint of qubits of a D-Wave QA.

In this article, we construct the coreset of an original dataset via *sparse variational inference* [6] and then employ this coreset for training the weighted SVM by using a D-Wave QA. Furthermore, we train the weighted SVM, posed as a QUBO problem, by using a D-Wave QA on the coreset instead of the original massive data, and we benchmark our classification results with respect to the *weighted* cSVM. As for practical and real-world EO data, we use *synthetic* data, *Iris* data, a *Hyperspectral Image* (HSI) of Indian Pine, and a *Polarimetric Synthetic Aperture Radar* (PolSAR) image of San Francisco characterized by its Stokes parameters [23].

Our paper is structured as follows: In Section 2, we present our datasets, and we construct the coresets of our datasets in Section 3. We introduce a *weighted* cSVM, and construct a *weighted* qSVM for our experiments in Section 4. Then, we train the *weighted* qSVM on our coresets by using a D-Wave QA and demonstrate our results with respect to the *weighted* cSVM in Section 5. Finally, we draw some conclusions in Section 6.

2. Our Datasets

We use four different datasets, namely *synthetic* data, *Iris* data, an Indian Pine *HSI*, and a *PolSAR* image of San Francisco characterized by its Stokes parameters [23,24]. The first two sets are used to understand the concept of a coreset, and the implementation of a weighted SVM on their coresets by using a D-Wave QA. Namely, we use the coresets of the first two to set the internal parameters of a D-Wave QA since the solutions generated by the D-Wave QA are affected by those internal parameters (called *annealing parameters*). The last two sets are employed as real-world EO data for constructing their coresets and for training the *weighted* qSVM on their coresets after the annealing parameters are set in a prior (see Figures 1 and 2). In the next sections, we use a notation “*weighted qSVM*” meaning that “training a *weighted* SVM posed as a QUBO problem by using a D-Wave QA”.

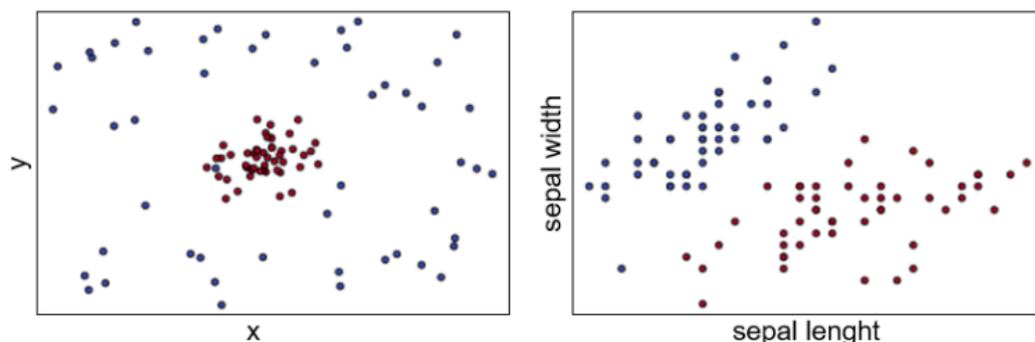


Figure 1. Synthetic data with two classes, and Iris data with two classes (Iris Setosa, and Iris Versicolour) characterized by two features (Sepal length, Sepal width).

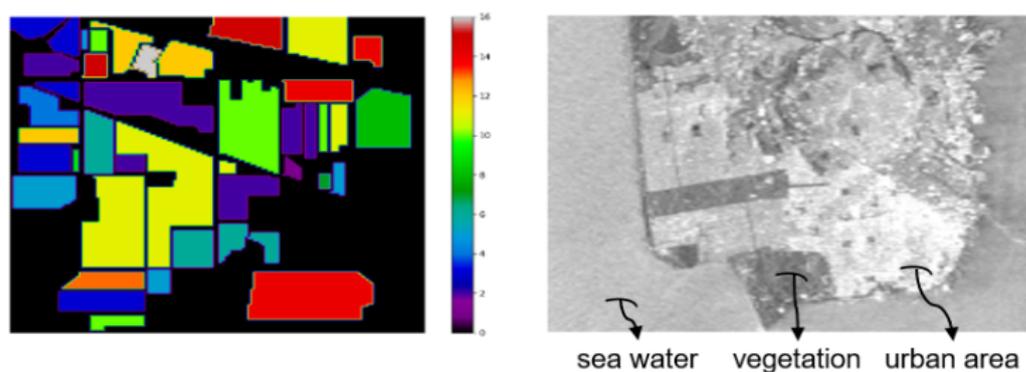


Figure 2. Indian Pine HSI with 16 classes {1: Alfalfa, 2: Corn-notill, 3: Corn-mintill, 4: Corn, 5: Grass-Pasture, 6: Grass-Trees, 7: Grass-Pasture-mowed, 8: Hay-windrowed, 9: Oats, 10: Soybean-notill, 11: Soybean-mintill, 12: Soybean-clean, 13: Wheat, 14: Woods, 15: Building-Grass-Drives, 16: Stones-Steel-Towers}, and PolSAR image of San Francisco with three classes.

2.1. Synthetic Data

We generated synthetic data with two classes (x_n, y_n) according to

$$x_n = r_n \begin{pmatrix} \cos \phi_n \\ \sin \phi_n \end{pmatrix} + \begin{pmatrix} \epsilon_n^x \\ \epsilon_n^y \end{pmatrix}, \quad y_n \in \{-1, +1\}, \tag{1}$$

where $r_n = 1$ if $y_n = -1$, and $r_n = 0.15$ if $y_n = +1$. ϕ_n is linearly spaced in $(0, 2\pi]$ for each class, and $\epsilon_n^x, \epsilon_n^y$ are samples drawn from a normal distribution with $\mu = 0, \sigma = 1$. We are replicating the data already demonstrated for training an SVM by using a D-Wave QA described in [25]. Moreover, we have $(x_n, y_n), n = 1, \dots, 100$ data points shown in Figure 1 (Left) and in Table 1.

2.2. Iris Data

Iris data consist of three classes (Iris Setosa, Iris Versicolour, and Iris Virginica), each of which has four features, namely sepal length, sepal width, petal length, and petal width. We consider a two-class dataset {Iris Setosa, Iris Versicolour} with a size of 100 data points, and each class is characterized by two features (sepal length, sepal width) shown in Figure 1 (Right) and Table 1.

Table 1. The two classes of Synthetic and Iris data.

	Synthetic Data	Iris Data
Classes	$\{-1, +1\}$	{setosa, versicolour}
Data size	100	100

2.3. Indian Pine HSI

An Indian Pine HSI obtained by the AVIRIS sensor comprises 16 classes; each class is characterized by 200 spectral bands (see Figure 2 (Left)). For simplicity, we use only two classes (see Table 2), and each class is characterized by two features instead of 200 spectral bands by exploiting Principal Component Analysis (PCA) [22].

Table 2. The two classes of the Indian Pine HSI; {1,2} represents {Alfalfa, Corn-notill}, {2,3} represents {Corn-notill, Corn-mintill}, etc.

Indian Pine HSI						
Classes	{1,2}	{2,3}	{3,4}	{4,5}	{5,6}	{6,7}
Data size	295	452	214	144	243	758

2.4. PolSAR Image of San Francisco

Each pixel of our PolSAR image is characterized by a 2×2 scattering matrix as follows:

$$S = \begin{pmatrix} s_{HH} & s_{HV} \\ s_{VH} & s_{VV} \end{pmatrix}, \tag{2}$$

where the first index of $s_{ij}, i, j \in \{H, V\}$ represents the polarization state of the incident polarized beam, and its second index represents the polarization state of the reflected polarized beam on targets. The off-diagonal elements of S are equal $s_{VH} = s_{HV}$ since our PolSAR image of San Francisco is a fully-polarized PolSAR image obtained by a monostatic radar [26,27].

The incident/reflected polarized beam can be represented by its complex amplitude in a polarization basis $\{\hat{H}, \hat{V}\}$ by

$$\vec{E}_0 = E_{H0}\hat{H} + E_{V0}\hat{V}. \tag{3}$$

The complex amplitude vector can be expressed by a so-called Jones vector

$$\vec{J} = \begin{pmatrix} E_{H0} \\ E_{V0} \end{pmatrix} = \begin{pmatrix} |E_{H0}|e^{i\phi_H} \\ |E_{V0}|e^{i\phi_V} \end{pmatrix}. \tag{4}$$

where ϕ_i are the phases of the polarized states. Furthermore, the scattering matrix S expressed in (2) is a mapping such that

$$S : \vec{J}_i \rightarrow \vec{J}_r, \quad \vec{J}_r = S\vec{J}_i, \tag{5}$$

where \vec{J}_i, \vec{J}_r is an incident and a reflected Jones vector, respectively. In matrix form, (5) can be re-expressed as

$$\begin{pmatrix} E_{H0}^r \\ E_{V0}^r \end{pmatrix} = \begin{pmatrix} s_{HH} & s_{HV} \\ s_{VH} & s_{VV} \end{pmatrix} \begin{pmatrix} E_{H0}^i \\ E_{V0}^i \end{pmatrix}. \tag{6}$$

The intensity of the reflected Jones vector is defined by

$$J = \begin{pmatrix} \langle E_{H0}^r E_{H0}^{r*} \rangle & \langle E_{H0}^r E_{V0}^{r*} \rangle \\ \langle E_{V0}^r E_{H0}^{r*} \rangle & \langle E_{V0}^r E_{V0}^{r*} \rangle \end{pmatrix} = \begin{pmatrix} J_{HH} & J_{HV} \\ J_{VH} & J_{VV} \end{pmatrix} \tag{7}$$

where $\langle \cdot \rangle$ stands for spatial averaging with a window size 7×7 pixels, and $*$ for conjugation. Furthermore, we can re-express this intensity by

$$\begin{pmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{pmatrix} = \begin{pmatrix} J_{HH} + J_{VV} \\ J_{HH} - J_{VV} \\ J_{VH} + J_{HV} \\ i(J_{HV} - J_{VH}) \end{pmatrix}, \tag{8}$$

where $q_1, q_2,$ and q_3 are called *Stokes vectors*. We normalize these *Stokes vectors* according to

$$q'_1 = \frac{q_1}{q_0}, \quad q'_2 = \frac{q_2}{q_0}, \quad q'_3 = \frac{q_3}{q_0}, \tag{9}$$

and the normalized $q'_1, q'_2,$ and q'_3 are called *Stokes parameters* [23].

Moreover, in this study, we use two classes for our PolSAR image of San Francisco, and the two classes are {urban area, sea water}, and {vegetation, sea water} shown in Figure 2 (Right) and in Table 3. In addition, each class is characterized by its Stokes parameters (q'_1, q'_2, q'_3) defined in (9).

Table 3. The two classes of our PolSAR image.

PolSAR Image of San Francisco		
Classes	{urban area, sea water}	{vegetation, sea water}
Data size	61,465	61,465

3. Coresets of Our Datasets

In Bayesian inference, a posterior density $p(\theta|x)$ is written for θ parameters and for $\{(x_i, t_i)\}_{i=1}^N$ data points with its labels t_i by

$$p(\theta|x) = \frac{1}{Z} \exp \left\{ \sum_{i=1}^N f_i(\theta) \right\} p_0(\theta), \tag{10}$$

where Z is a partition function, $f_i(\theta)$ is a potential function, and $p_0(\theta)$ is a prior. For big massive data, the estimation of the posterior distribution is intractable, and hence, in practice, a Markov Chain Monte Carlo (MCMC) method is widely used to obtain samples from the posterior expressed by (10) [28].

To reduce the computational time of an MCMC method, the authors of [5–7] proposed to run the MCMC method on a small, weighted subset (i.e., coreset) of big massive data. They derived a sparse vector of nonnegative weights w such that only $M \ll N$ are non-zero, where M is the size of a coreset. Furthermore, the authors proposed to approximate the weighted posterior distribution and run the MCMC method on the approximate distribution as follows:

$$p_w = p_w(\theta|x) = \frac{1}{Z(w)} \exp \left\{ \sum_{i=1}^N w_i f_i(\theta) \right\} p_0(\theta). \tag{11}$$

We denote the full distribution of an original big massive dataset as $p_1 = p_1(\theta|x)$. More importantly, this posterior becomes tractable.

For assembling the coresets of our datasets presented in Table 1–3, we use an algorithm via *sparse variational inference* for finding the sparse vector of nonnegative weights w and for approximating the posterior distribution (11) proposed by [6]. Here, the sparse vector of nonnegative weights w is found by optimizing a *sparse variational inference* problem:

$$\hat{w} = \min_w D_{KL}(p_w||p_1) \quad s.t. \quad w \geq 0, \quad \|w\|_0 \leq M, \tag{12}$$

where \hat{w} is an optimal sparse vector weight, and $D_{KL}(p_w||p_1)$ is the Kullback–Leibler (KL) divergence which measures the similarity between two distributions (smaller is better):

$$D_{KL}(p_w||p_1) = \sum_x p_w \log \frac{p_w}{p_1}. \tag{13}$$

Moreover, the smaller value of the KL divergence implies that we can estimate the parameters θ in (11) by using a coreset yielding similar results with respect to the ones in (10) by using its original massive dataset.

We derived the optimal sparse vector weights \hat{w} and the coresets of our dataset such that

$$\{(x_i, t_i)\}_{i=1}^N \rightarrow \{(c_i, t_i, \hat{w}_i)\}_{i=1}^M, \quad \hat{w}_i \in R_{\geq 0}, \quad (14)$$

where (x_i, t_i) represents an original dataset, while (c_i, t_i, \hat{w}_i) is our newly assembled coreset. In addition, we computed the similarity between our datasets and the corresponding coresets by using their KL divergences (see Table 4). Our results show that our coresets are very small in size compared with our original datasets, and the KL divergences between the original dataset and our coresets are comparatively small in most instances.

Table 4. Coresets of our datasets presented in Table 1–3, and the closeness between the original dataset and its coreset is measured by KL divergence.

Classes	Data Size	Coreset Size	KL Divergence
{−1, +1}	100	20	0.008194
{setosa, versicolour}	100	22	0.053002
{1, 2}	295	79	0.573451
{2, 3}	452	56	0.003121
{3, 4}	214	33	0.000600
{4, 5}	144	41	0.017201
{5, 6}	243	41	0.001823
{6, 7}	758	125	0.492636
{urban area, sea water}	61,465	501	0.125072
{vegetation, sea water}	61,465	343	0.272749

4. Weighted Classical and Quantum SVMs on Our Coresets

4.1. Weighted Classical SVMs

In the previous section, we assembled the coreset of our original datasets shown in Table 4 as

$$\{(c_i, t_i, \hat{w}_i)\}_{i=1}^M, \quad c_i \in R^2, \quad \hat{w}_i \in R_{\geq 0}. \quad (15)$$

To train a weighted SVM for our coresets represented via (15) by using a conventional computer, we express a weighted SVM as

$$\begin{aligned} \text{minimize} \quad & H(\alpha) = \frac{1}{2} \sum_{ij} \alpha_i \alpha_j t_i t_j k(c_i, c_j) - \sum_i \alpha_i \\ \text{subject to} \quad & 0 \leq \alpha_i \leq C_i, \\ \text{and} \quad & \sum_i \alpha_i t_i = 0, \quad \alpha_i \in R, \end{aligned} \quad (16)$$

where $C_i = \hat{w}_i C$ is a regularization parameter, and $k(\cdot, \cdot)$ is the kernel function of the SVM [28]. This formulation of the SVM is called a *weighted* cSVM [29]; sometimes, it is called a kernel-based *weighted* cSVM. The point c_i with $\alpha_i \neq 0$ is called a support vector.

After training the *weighted* cSVM, for a given test point $x_t \in R^2$, the decision function for its class label is defined by:

$$\hat{t} = \text{sign}[f(x_t)] = \text{sign} \left[\sum_i \alpha_i t_i k(c_i, x_t) + b \right], \quad (17)$$

where $\text{sign}(f(x_t)) = 1$ if $f(x_t) > 0$, $\text{sign}(f(x_t)) = -1$ if $\text{sign}(f(x_t)) < 0$, and $\text{sign}(f(x_t)) = 0$ otherwise. The decision boundary is an optimum hyperplane drawn by data points such that $f(x_t) = 0$ [28]. The bias b is expressed following [25]:

$$b = \frac{\sum_i \alpha_i (C_i - \alpha_i) \left[t_i - \sum_j \alpha_j t_j k(c_j, c_i) \right]}{\sum_i \alpha_i (C_i - \alpha_i)}. \quad (18)$$

The kernel-based *weighted* cSVM is a powerful technique since the kernel function maps non-separable features to higher dimensional separable features, and the decision boundary is less sensitive to outliers due to the weighted constraints C_i [25,29]. Furthermore, the choice of the kernel function has a huge impact on the decision boundary, and the types of the kernel function are linear, polynomial, Matern, and a radial basis function (rbf) [28]. A widely-used kernel is an rbf defined by

$$\text{rbf}(c_i, c_j) = \exp\left\{-\gamma \|c_i - c_j\|^2\right\}, \quad (19)$$

where $\gamma > 0$ is a parameter.

4.2. Weighted Quantum SVMs

A weighted quantum SVM (in short, *weighted* qSVM) is the training result of the *weighted* cSVM given in (16) on a D-Wave QA. The D-Wave QA is a quantum annealer with a specific *Pegasus* topology for the interaction of its qubits, and it is specially designed to solve a QUBO problem:

$$H(\mathbf{z}) = \sum_{i,j} z_i Q_{ij} z_j, \quad z_i, z_j \in \{0, 1\}, \quad (20)$$

where z_i, z_j are called logical variables, and Q_{ij} includes a bias term h_i and the interaction strength of the logical variables g_{ij} [19]. Physical states of the *Pegasus* topology are called physical variables, two-state qubits residing at the edges of the *Pegasus* topology; a QUBO problem is also called a *problem energy*. The D-Wave QA anneals (evolves) from an initial to its final energy (problem energy) according to

$$H(T) = (1 - \varepsilon(T))H_0 + \varepsilon(T)H(\mathbf{z}), \quad (21)$$

where H_0 is an initial energy, T is the annealing time in microseconds, and $\varepsilon(T)$ is an annealing parameter in the range of $[0, 1]$.

Furthermore, to train the *weighted* qSVM on our coresets by using a D-Wave QA, we pose the *weighted* cSVM with an rbf kernel expressed by (16) and (19) as a QUBO problem. Here, we duplicate the formulation for posing the *weighted* cSVM as a QUBO problem in the article [25].

The variables of the *weighted* cSVM are decimal integers when the ones of the QUBO problem are binaries. Hence, we use a one-hot encoding form for the variables of the *weighted* cSVM

$$\alpha_i = \sum_{k=0}^{K-1} B^k z_{Ki+k}, \quad z_{Ki+k} \in \{0, +1\} \quad (22)$$

where K is the number of binary variables (bits), and B is the base. We insert this one-hot encoding form into the *weighted* cSVM given in (16), and formulate the second constraint of (16) as a squared penalty term

$$\left(\sum_i \alpha_i t_i \right)^2 = 0 \quad \rightarrow \quad \left(\sum_{ik} B^k z_{Ki+k} t_i \right)^2 = 0. \quad (23)$$

By using a Lagrange multiplier λ , we transform our *weighted* cSVM into the QUBO problem (20)

$$\begin{aligned} \text{minimize } H(z) &= \frac{1}{2} \sum_{ijkl} z_{Ki+k} z_{Kj+l} B^{k+l} t_i t_j k(c_i, c_j) \\ &\quad - \sum_{ik} B^k z_{Ki+k} + \lambda \left(\sum_{ik} B^k z_{Ki+k} t_i \right)^2 \\ &= \sum_{ij} \sum_{kl} z_{Ki+k} Q_{Ki+k, Kj+l} z_{Kj+l}, \end{aligned} \quad (24)$$

where

$$Q_{Ki+k, Kj+l} = \frac{1}{2} B^{k+l} t_i t_j (k(c_i, c_j) + \lambda) - \delta_{ij} \delta_{kl} B^k. \quad (25)$$

Note that the first constraint of (16) is satisfied automatically since the one-hot encoding form given in (22) is always greater than zero, and hence the maximum value for each α_i is given by

$$C_i = \hat{w}_i \sum_{k=1}^K B^k. \quad (26)$$

For training the *weighted* qSVM, we concentrated on four hyperparameters which are the parameter γ of the RBF expressed by (19), the number of binary bits K , the base B , and the Lagrange multiplier λ given in (24); thus, we used the hyperparameters (γ, K, B, λ) . For our applications, we set these hyperparameters to $(\gamma = 16, K = 2, B = 2, \lambda = 0)$ as proposed by [25] since these settings of the hyperparameters for the *weighted* qSVM generate competitive results with the ones generated by the *weighted* cSVM. For setting the bias defined in (18), we used the *weighted* cSVM.

In the next section, we train the *weighted* cSVM given in (16) and the *weighted* qSVM expressed by (24) on our coresets (see Table 4). In addition, we demonstrate how to program a D-Wave QA for obtaining a good solution of (24) since the solutions yielded by a D-Wave QA are greatly dependent upon the *embedding* of the logical variables into their corresponding physical variables, and the *annealing parameters* (annealing time, number of reads, and chain strength) [30].

5. Our Experiments

In our experiments, we trained our *weighted* cSVM and our *weighted* qSVM (models) on the coresets, and we tested our models on the original datasets (see Table 4). In addition, we set the hyperparameters of our *weighted* qSVM to $(\gamma = 16, K = 2, B = 2, \lambda = 0)$, and for training (i.e., for setting of the bias) of the *weighted* cSVM, we used the Python module *scikit-learn* [31].

For defining the *annealing parameters* (annealing time, number of reads, and chain strength) of a D-Wave QA, we first ran quantum experiments on *synthetic* two-class data, and *Iris* data. Then, by leveraging these *annealing parameters*, we used our real-world EO data (the Indian Pine HSI and the PolSAR image of San Francisco) for evaluating our proposed method, “by training the *weighted* qSVM on the coreset of our EO data instead of a massive original EO data due to the small quantum computer (D-Wave QA) with only few qubits”.

5.1. Synthetic Two-Class Data and Iris Data

For training the *weighted* qSVM expressed by (24), we first experimented on our coresets of *synthetic* two-class data and *Iris* data shown in Table 4 in order to optimize the *annealing parameters* (annealing time, number of reads, and chain strength) of a D-Wave QA. In addition, we benchmarked the classification results generated by the *weighted* qSVM compared with the *weighted* cSVM. This had the advantage that we could easily

tune the *annealing parameters* and visualize the generated results, both for quantum and classic settings.

In Figure 3 (in Table 5), we show our results for the classification of *synthetic two-class data* and *Iris data*. Our results demonstrate that the *weighted qSVM* performs well in comparison with the *weighted cSVM* for both coresets (often better for *Iris data*).

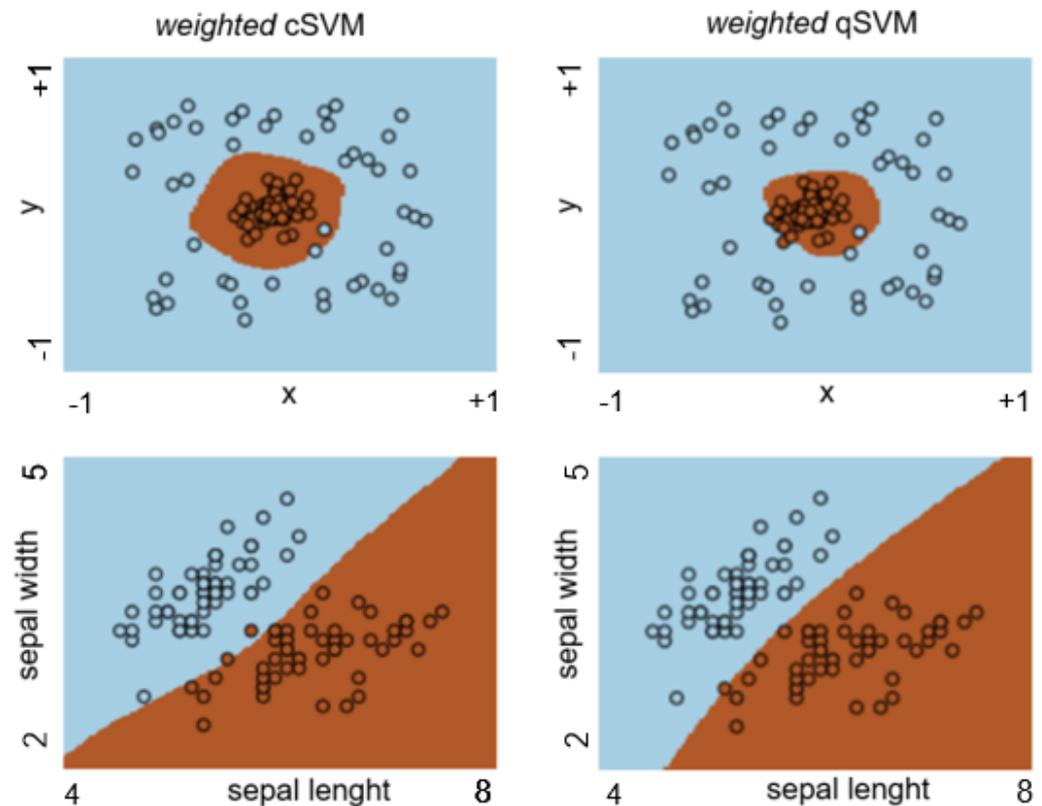


Figure 3. Top: *Synthetic two data*; Bottom: *Iris data*. The visual results of our experiments generated by the *weighted cSVM* given in (16) and *weighted qSVM* expressed by (24). Our visual results demonstrate that our *weighted qSVM* generalizes the decision boundary of a given dataset better than its counterpart *weighted cSVM*.

To obtain these good solutions generated by our *weighted qSVM*, we set the annealing parameters of the D-Wave QA as follows:

- Annealing time: We controlled the annealing time by an *anneal schedule*. The anneal schedule is defined by the four series of pairs $[T, \epsilon(T)]$ defined in (21). We set the annealing schedule accordingly: $[T, \epsilon(T)] \in \{[0.0, 0.0], [1.0, 0.40], [19.0, 0.40], [20.0, 1.0]\}$.
- Number of reads: 10,000
- Chain strength: 50.

Table 5. The classification accuracy of the weighted quantum svm (in short, qacc), and the weighted classical svm (in short, cacc) on our coresets.

Classes	Coreset Size	Qacc	Cacc
{−1, +1}	20	0.95	0.97
{setosa, versicolour}	22	0.99	0.98
{1, 2}	79	0.96	0.96
{2, 3}	56	0.70	0.70
{3, 4}	33	0.88	0.88
{4, 5}	41	0.78	0.78
{5, 6}	41	0.71	0.71
{6, 7}	125	0.92	0.90
{urban area, sea water}	501	0.99	0.98
{vegetation, sea water}	343	0.99	0.99

5.2. Indian Pine HSI and PolSAR Image of San Francisco

As real-world EO data, we used the coresets of an Indian Pine HSI, and a PolSAR image of San Francisco for training the *weighted* qSVM when setting the annealing parameters of a D-Wave QA set as described above. Initially, we ran a number of quantum experiments on our coresets. In Table 5, we show the classification accuracy of our *weighted* qSVM results in comparison with the ones yielded by the *weighted* cSVM.

Our results explicitly demonstrate that the coresets obtained via *sparse variational inference* are small and representative subsets of our original datasets validated by their KL divergences shown in Table 4. In addition, our *weighted* qSVM generates its decision results with the same classification accuracy as for the *weighted* cSVM; in some instances, the *weighted* qSVM outperforms the *weighted* cSVM. Furthermore, by exploiting the coresets, we reduced the computational time of training with the *weighted* qSVM and the MCMC method for inferring the parameters of the posterior distribution as *proved* theoretically and *demonstrated* experimentally in [5,6].

6. Discussion and Conclusions

Quantum algorithms (e.g., Grover’s search algorithm) are designed to process data in quantum computers, and they are known to achieve quantum advantages over their conventional counterparts. Motivated by these quantum advantages, quantum computers based on quantum information science are being built for solving some problems (or running some algorithms) more efficiently than a conventional computer. However, currently available quantum computers (a D-Wave quantum annealer, and a gate-based quantum computer) are very small in input quantum bits (qubits). A very specific type of a quantum computer is a D-Wave quantum annealer (QA); it is designed to solve a Quadratic Unconstrained Binary Optimization (QUBO) problem belonging to a family of quadratic programming problems better than conventional methods.

For Earth observation, satellite images obtained from aircraft or satellite platforms are massive and represent hard heterogeneous data to train ML models on a conventional computer. As a practical and real-world EO dataset, we used *synthetic* data, *Iris* data, a *Hyperspectral Image* (HSI) of Indian Pine, and a *Polarimetric Synthetic Aperture Radar* (PolSAR) image of San Francisco. One of the well-known methods in ML is a Support Vector Machine (SVM): This represents a quadratic programming problem. A global minimum of such a problem can be found by employing a classical method. However, its quadratic form allows us to use a D-Wave QA for finding the solution of an SVM better than a conventional computer. Thus, we can pose an SVM as a QUBO problem, and we named an SVM-to-QUBO transformation as a quantum SVM (qSVM). Then, we can train the qSVM on our real-world EO data by using a D-Wave QA. However, the number of the physical variables

of the qSVM is much larger than that of the logical variables of a D-Wave QA due to the massive EO data and the very few qubits.

Therefore, in our paper, we employed the *coreset* ("core of a dataset") concept via *sparse variational inference*, where the coreset is a very small and representative weighted subset of the original dataset. By assembling and exploiting the coreset of *synthetic* data and *Iris* data shown in Table 4, we trained a *weighted* qSVM posed as a QUBO problem on these coresets in order to set the *annealing parameters* of a D-Wave QA. We then presented our obtained visual results and the classification accuracy of *synthetic* and *Iris* data in Figure 2 and in Table 5, respectively, in contrast to the ones of the *weighted* cSVM. Our results show that the *weighted* qSVM is competitive in comparison with the *weighted* cSVM – and for *Iris* data even better than the *weighted* cSVM.

Finally, we assembled the coresets of our real-world EO data (from an *HSI* of Indian Pine, and a *PolSAR* image of San Francisco), and demonstrated the similarity between our real-world EO data and its coreset by analyzing their KL divergence. The KL divergence test proved that our coresets are valid, small, and representative weighted subsets of our real-world EO data (see Table 4). Then, we trained the *weighted* qSVM on our coresets by using a D-Wave QA to prove that our *weighted* qSVM generates classification results being competitive with the *weighted* cSVM in Table 5. The annealing parameters of the D-Wave QA were already defined in the prior section. In some instances, one can see that our *weighted* qSVM outperforms the *weighted* cSVM.

As ongoing and future work, we intend to develop a novel method for assembling coresets with balanced labels via *sparse variational inference* since currently available techniques generate unbalanced labels. Furthermore, we plan to design hybrid quantum-classical methods for different real-world EO problems. These hybrid quantum-classical methods will perform a dimensionality-reduction of remotely-sensed images (in the spatial-dimension) by using our established methods, and will reduce the size of our training/test data by using a coreset generating balanced labels when we process these small datasets on a small quantum computer.

Author Contributions: Conceptualization, S.O.; methodology, S.O.; software, S.O.; investigation, S.O.; resources, M.D.; data curation, S.O.; writing—original draft preparation, S.O.; writing—review and editing, S.O.; supervision, M.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research has no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The *hyperspectral* image of Indian Pine is available online: <https://aviris.jpl.nasa.gov/>, accessed on 30 August 2021. The code for a coreset construction via *sparse variational inference* is available in: <https://github.com/trevorcampbell/bayesian-coresets>, accessed on 30 August 2021, and the code for a quantum Support Vector Machine is available: https://gitlab.jsc.fz-juelich.de/cavallaro1/quantum_svm_algorithms accessed on 30 August 2021.

Acknowledgments: We would like to greatly acknowledge Gottfried Schwarz (DLR, Oberpfaffenhofen) for his valuable comments and contributions for enhancing the quality of our paper. In addition, the authors gratefully acknowledge the Juelich Supercomputing Centre (<https://www.fzjuelich.de/ias/jsc>, accessed on 30 August 2021) for funding this project by providing computing time through the Juelich UNified Infrastructure for Quantum computing (JUNIQ) on a D-Wave quantum annealer.

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

References

1. Richards, J. *Remote Sensing Digital Image Analysis*; Springer: Berlin/Heidelberg, Germany, 2013. [[CrossRef](#)]
2. Cheng, G.; Xie, X.; Han, J.; Guo, L.; Xia, G.S. Remote Sensing Image Scene Classification Meets Deep Learning: Challenges, Methods, Benchmarks, and Opportunities. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 3735–3756. [[CrossRef](#)]
3. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)]
4. Marmanis, D.; Datcu, M.; Esch, T.; Stilla, U. Deep Learning Earth Observation Classification Using ImageNet Pretrained Networks. *IEEE Geosci. Remote Sens. Lett.* **2016**, *13*, 105–109. [[CrossRef](#)]
5. Manousakas, D.; Xu, Z.; Mascolo, C.; Campbell, T. Bayesian Pseudocoresets. In *Advances in Neural Information Processing Systems*; Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M.F., Lin, H., Eds.; Curran Associates, Inc.: New York, NY, USA, 2020; Volume 33, pp. 14950–14960.
6. Campbell, T.; Beronov, B. Sparse Variational Inference: Bayesian Coresets from Scratch. In *Advances in Neural Information Processing Systems*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: New York, NY, USA, 2019; Volume 32.
7. Campbell, T.; Broderick, T. Bayesian Coreset Construction via Greedy Iterative Geodesic Ascent. *arXiv* **2018**, arXiv:1802.01737.
8. Tukan, M.; Baykal, C.; Feldman, D.; Rus, D. On Coresets for Support Vector Machines. *arXiv* **2020**, arXiv:2002.06469.
9. Harrow, A.W. Small quantum computers and large classical data sets. *arXiv* **2020**, arXiv:2004.00026.
10. Tomesh, T.; Gokhale, P.; Anschuetz, E.R.; Chong, F.T. Coreset Clustering on Small Quantum Computers. *Electronics* **2021**, *10*, 1690. [[CrossRef](#)]
11. Preskill, J. Quantum Computing in the NISQ era and beyond. *Quantum* **2018**, *2*, 79. [[CrossRef](#)]
12. Denchev, V.S.; Boixo, S.; Isakov, S.V.; Ding, N.; Babbush, R.; Smelyanskiy, V.; Martinis, J.; Neven, H. What is the Computational Value of Finite-Range Tunneling? *Phys. Rev. X* **2016**, *6*, 031015. [[CrossRef](#)]
13. Dunjko, V.; Briegel, H.J. Machine learning & artificial intelligence in the quantum domain: a review of recent progress. *Rep. Prog. Phys.* **2018**, *81*, 074001. [[CrossRef](#)]
14. Rebertrost, P.; Mohseni, M.; Lloyd, S. Quantum Support Vector Machine for Big Data Classification. *Phys. Rev. Lett.* **2014**, *113*, 130503. [[CrossRef](#)]
15. Huang, H.Y.; Broughton, M.; Mohseni, M.; Babbush, R.; Boixo, S.; Neven, H.; McClean, J.R. Power of data in quantum machine learning. *Nat. Commun.* **2021**, *12*, 2631. [[CrossRef](#)]
16. Biamonte, J.; Wittek, P.; Pancotti, N.; Rebertrost, P.; Wiebe, N.; Lloyd, S. Quantum machine learning. *Nature* **2017**, *549*, 195–202. [[CrossRef](#)]
17. Otgonbaatar, S.; Datcu, M. Natural Embedding of the Stokes Parameters of Polarimetric Synthetic Aperture Radar Images in a Gate-Based Quantum Computer. *IEEE Trans. Geosci. Remote Sens.* **2021**, 1–8. [[CrossRef](#)]
18. Otgonbaatar, S.; Datcu, M. Classification of Remote Sensing Images with Parameterized Quantum Gates. *IEEE Geosci. Remote Sens. Lett.* **2021**, 1–5. [[CrossRef](#)]
19. Boothby, K.; Bunyk, P.; Raymond, J.; Roy, A. Next-Generation Topology of D-Wave Quantum Processors. *arXiv* **2020**, arXiv:2003.00133.
20. Farhi, E.; Goldstone, J.; Gutmann, S.; Sipser, M. Quantum Computation by Adiabatic Evolution. *arXiv* **2000**, arXiv:0001106.
21. Cavallaro, G.; Willsch, D.; Willsch, M.; Michielsen, K.; Riedel, M. Approaching Remote Sensing Image Classification with Ensembles of Support Vector Machines on the D-Wave Quantum Annealer. In Proceedings of the IGARSS 2020–2020 IEEE International Geoscience and Remote Sensing Symposium, Waikoloa, HI, USA, 26 September–2 October 2020; pp. 1973–1976. [[CrossRef](#)]
22. Otgonbaatar, S.; Datcu, M. A Quantum Annealer for Subset Feature Selection and the Classification of Hyperspectral Images. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 7057–7065. [[CrossRef](#)]
23. Shang, F.; Hirose, A. Quaternion Neural-Network-Based PolSAR Land Classification in Poincare-Sphere-Parameter Space. *IEEE Trans. Geosci. Remote Sens.* **2014**, *52*, 5693–5703. [[CrossRef](#)]
24. Cloude, S.R.; Papathanassiou, K.P. Polarimetric SAR interferometry. *IEEE Trans. Geosci. Remote Sens.* **1998**, *36*, 1551–1565. [[CrossRef](#)]
25. Willsch, D.; Willsch, M.; De Raedt, H.; Michielsen, K. Support vector machines on the D-Wave quantum annealer. *Comput. Phys. Commun.* **2020**, *248*, 107006. [[CrossRef](#)]
26. Lee, J.S.; Boerner, W.M.; Schuler, D.; Ainsworth, T.; Hajnsek, I.; Papathanassiou, K.; Lüneburg, E. A Review of Polarimetric SAR Algorithms and their Applications. *J. Photogramm. Remote Sens. China* **2004**, *9*, 31–80.
27. Moreira, A.; Prats-Iraola, P.; Younis, M.; Krieger, G.; Hajnsek, I.; Papathanassiou, K. A Tutorial on Synthetic Aperture Radar. *IEEE Geosci. Remote Sens. Mag. (GRSM)* **2013**, *1*, 6–43. [[CrossRef](#)]
28. Sheykhmousa, M.; Mahdianpari, M.; Ghanbari, H.; Mohammadimanesh, F.; Ghamisi, P.; Homayouni, S. Support Vector Machine Versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2020**, *13*, 6308–6325. [[CrossRef](#)]
29. Yang, X.; Song, Q.; Cao, A. Weighted support vector machine for data classification. In Proceedings of the 2005 IEEE International Joint Conference on Neural Networks, Montreal, QC, Canada, 31 July–4 August 2005; Volume 2, pp. 859–864. [[CrossRef](#)]

-
30. Otgonbaatar, S.; Datcu, M. Quantum annealer for network flow minimization in InSAR images. In Proceedings of the EUSAR 2021: 13th European Conference on Synthetic Aperture Radar, Online, 29 March–1 April 2021; pp. 1–4.
 31. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.

5

Main Results II: Toward Quantum Advantage

Toward gaining quantum advantage using a future quantum computer, the prior chapter presents real-world applications to gain insight into programming faulty quantum computers comprising NISQ computers and QA devices and to recognize their advantage and imperfection compared with conventional classical computers. Chapter 4 of this dissertation demonstrates how classical data points can be embedded in faulty quantum computers. We also compared the performance of quantum approaches with traditional classical methods. We aimed to gain insight into how to program a quantum computer with error-prone qubits and quantum gates. Until now, the following quote guides us, “We have to learn to walk before we can run.”

In order to achieve quantum advantage with future quantum computers having error-free qubits and quantum gates, we need to benchmark the performance of quantum approaches and estimate the quantum resources required for solving computational problems.

5.1 The Quantum Computing Assessment

This chapter provides the challenges and assessment of existing and future quantum computers for tackling notoriously hard computational problems. We propose a classical method for benchmarking quantum approaches, and evaluating for different quantum computers based on their errors and size. We also propose how to profit from a quantum computer integrated into High-Performance Computing (HPC) systems. In conclusion, we emphasize, “If an error is corrected whenever it is recognized as such, the path of error is the path of truth” by Hans Reichenbach. This chapter is based on three publications:

- **G:** S. Otgonbaatar and D. Kranzlmüller, “Quantum-inspired Tensor Network for Earth Science,” *IEEE International Geoscience and Remote Sensing Symposium*, Pasadena, CA, USA, 2023, pp. 788-791, doi: 10.1109/IGARSS52108.2023.10282577 [73].
- **H:** S. Otgonbaatar and D. Kranzlmüller, “Exploiting the Quantum Advantage for Satellite Image Processing: Review and Assessment,” in *IEEE Transactions on Quantum Engineering*, vol. 5, pp. 1-9, 2024, art no. 3100309, doi: 10.1109/TQE.2023.3338970 [72].
- **I:** S. Otgonbaatar, et al., “Quantum Computing for Climate Change Detection, Climate Modeling, and Climate Digital Twins,” in preparation for *Nature Reviews Earth and Environment* [74].

5.1.1 Publication G: Quantum-inspired Tensor Network for Earth Science

S. Otgonbaatar and D. Kranzlmüller

Abstract. DL is one of the successful methodologies that help extract informative patterns and insights from large-scale datasets, such as satellite images. However, it consists of thousands to millions of training parameters, which need a lot of electrical power to analyze large datasets, making them computationally expensive. We propose to use a quantum-inspired tensor network to compress the trainable parameters of [Physics-Informed Neural Networks \(PINNs\)](#) in Earth science. PINNs are DL models that enforce the law of physics, embedding them in the DL model. Additionally, we use tensor decomposition to improve the spectral resolution of hyperspectral images. A quantum-inspired tensor network is also an efficient way to represent and benchmark QML models on big datasets on [Graphics Processing Unit \(GPU\)](#) tensor cores depending on the entanglement. We offer two key contributions. Firstly, we reduce the number of trainable parameters of PINNs using a quantum-inspired tensor network. Secondly, we improve the spectral resolution of satellite images by employing tensor decomposition. We use Burger's equation as a benchmark [Partial Differential Equation \(PDE\)](#). As practical satellite data, we use hyperspectral images of Indian Pine, USA and Pavia University, Italy. Our results demonstrate the potential of quantum-inspired tensor networks and tensor decomposition to improve the efficiency, accuracy, and practicality of DL models and Earth observation applications.

QUANTUM-INSPIRED TENSOR NETWORK FOR EARTH SCIENCE

Soronzonbold Otgonbaatar, Dieter Kranzlmüller

German Aerospace Center (DLR), Ludwig-Maximilians-Universität München

ABSTRACT

Deep Learning (DL) is one of many successful methodologies to extract informative patterns and insights from ever increasing noisy large-scale datasets (in our case, satellite images). However, DL models consist of a few thousand to millions of training parameters, and these training parameters require tremendous amount of electrical power for extracting informative patterns from noisy large-scale datasets (e.g., computationally expensive). Hence, we employ a quantum-inspired tensor network for compressing trainable parameters of physics-informed neural networks (PINNs) in Earth science. PINNs are DL models penalized by enforcing the law of physics; in particular, the law of physics is embedded in DL models. In addition, we apply tensor decomposition to HyperSpectral Images (HSIs) to improve their spectral resolution. A quantum-inspired tensor network is also the native formulation to efficiently represent and train quantum machine learning models on big datasets on GPU tensor cores. Furthermore, the key contribution of this paper is twofold: (I) we reduced a number of trainable parameters of PINNs by using a quantum-inspired tensor network, and (II) we improved the spectral resolution of remotely-sensed images by employing tensor decomposition. As a benchmark PDE, we solved Burger's equation. As practical satellite data, we employed HSIs of Indian Pine, USA and of Pavia University, Italy.

Index Terms— Tensor decomposition, quantum-inspired tensor decomposition, quantum-inspired machine learning.

1. INTRODUCTION

Deep Learning (DL) is a machinery for extracting most informative patterns, insights from large-scale data, and apply this knowledge to make predictions [1]. DL models currently have been outperforming conventional techniques and methods in science and engineering, even in remote sensing and Earth science [2, 3, 4]. However, DL models compose of a huge number of parameters, making their interpretation and predictions on large-scale data difficult. Their energy requirements also extremely limit their scalability (or computationally expensive) [5]. Hence, the authors of the articles [6, 7, 8] utilized a quantum-inspired tensor network to compress the parameters (e.g., hidden layers) of DL models and to decompose data tensors in very small factor matrices. Here, ten-

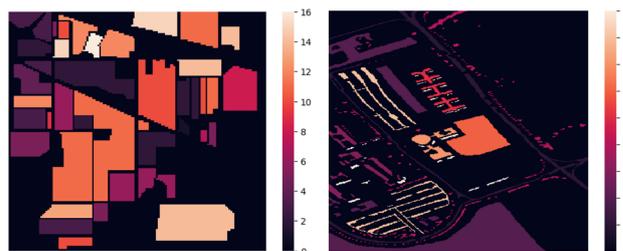


Fig. 1. Satellite datasets: [Left] HSIs of Indian Pine, USA and [Right] of Pavia University, Italy

sors are multidimensional arrays which can generalize vectors and matrices. A quantum-inspired tensor network can compress the training parameters of DL models and decompose data tensors in a small number of factor matrices. It is also widely used to represent quantum Machine Learning models as tensor-networks, which can be efficiently trained on big real-world datasets on GPU tensor cores [9].

Physics-Informed Neural Networks (PINNs) are DL models (e.g., Neural Networks), whose training parameters are penalized by enforcing the law of physics [10]; namely, the law of physics is embedded in Neural Networks (NNs). Moreover, PINNs can be utilized to compute and analyse computationally expensive Partial Differential Equations (PDEs) when data is of limited quantity and quality [11]. However, PINNs are still computationally expensive for obtaining solutions to PDEs in Earth science.

Remotely-sensed datasets are data tensors $\mathcal{X} \in \mathbb{R}^{I_1 \times \dots \times I_n}$ which are so complex and diverse that they cannot be easily classified and analyzed even by using DL models. In particular, these datasets are characterized by not only volume but also another so-called “4V” features (Volume, Variety, Veracity, and Velocity) [12].

The key contribution of this paper is twofold: The first contribution of this paper is that we reduced a number of trainable parameters of DL models (i.e. PINNs) by using the quantum-inspired tensor network [13]. The compressed DL models can be also applied to analyse and classify big real-world datasets as shown in the article [7]. The second contribution of this paper is that we improved the spectral resolution

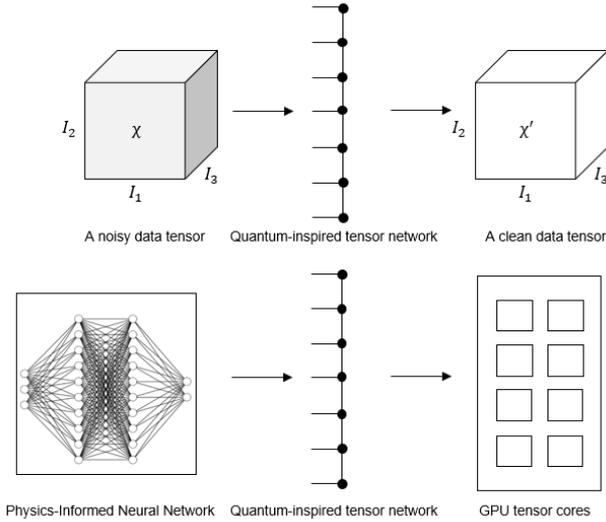


Fig. 2. The two contributions of this paper in pictorial representation: [Top] Quantum-inspired tensor network (decomposition) for improving spectral resolution of real-world noisy data tensor, and [Bottom] Quantum-inspired tensor network for compressing Physics-Informed Neural Networks, which can be efficiently simulated on GPU tensor cores.

of remotely-sensed images by employing tensor decomposition. As practical satellite data, we employed HSIs of Indian Pine, USA and of Pavia University, Italy. As a PDE, we considered Burger’s equation.

2. OUR DATASETS

We use practical satellite datasets and refer them as 3rd-order data tensors. In particular, the HSI of Indian Pine is the data tensor $\mathbb{R}^{240 \times 240 \times 200}$ with 16 classes, and the HSI of Pavia University is the data tensor $\mathbb{R}^{610 \times 340 \times 103}$ with 9 classes (see Fig. 1).

3. OUR METHODOLOGY

Remotely sensed images can be viewed as 3rd-order data tensors $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$. The 3rd-order data tensors can be decomposed in factor matrices by using so-called CANDECOMP/PARAFAC (CP)-decomposition [6]:

$$\mathcal{X} = \sum_{r=1}^R \mathbf{a}_r \circ \mathbf{b}_r \circ \mathbf{c}_r, \quad (1)$$

where R , called the rank, is a real positive number, “ \circ ” denotes an outer product, and $\mathbf{a}_r \in \mathbb{R}^{I_1}$, $\mathbf{b}_r \in \mathbb{R}^{I_2}$, and $\mathbf{c}_r \in \mathbb{R}^{I_3}$ are factor matrices (see Fig. 2 [Top]).

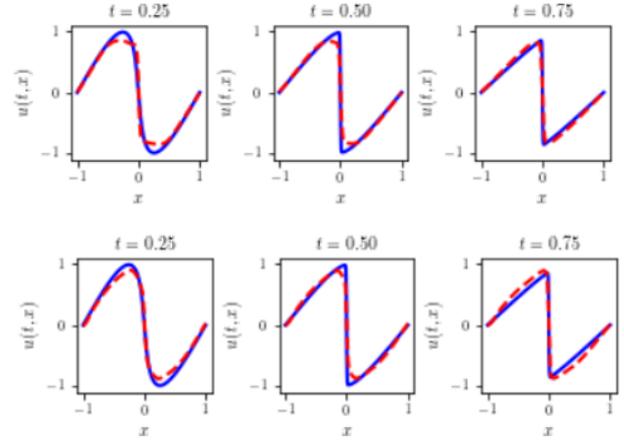


Fig. 3. A solution to Burger’s equation (blue is an exact solution, and red is a predicted solution): [Top] The original PINN, and [Bottom] The compressed PINN

Another commonly used quantum-inspired tensor network is Tensor Train (TT)-decomposition, called also Matrix Product State (MPS) in quantum physics [14]. TT-decomposition expresses a 3rd-order tensor as core tensors and factor matrices:

$$\mathcal{X} = \mathbf{A} \times_3^1 \mathbf{G}^{(2)} \times_3^1 \mathbf{B}, \quad (2)$$

where $\mathbf{G}^{(2)} \in \mathbb{R}^{R_1 \times I_2 \times R_2}$ is a core tensor, \mathbf{A} and \mathbf{B} are factor matrices, and \times_3^1 is called a mode-(k,l) product.

TT-decomposition can compress DL models, and the compressed DL models can generate classes with the similar accuracy as their non-compressed ones [7] (see Fig. 2 [Bottom]). In addition, TT-decomposition is widely employed to efficiently simulate quantum circuits on conventional computers. Hence, TT-decomposition have been applying to design and train quantum-inspired machine learning models on large-scale datasets on GPU tensor cores [15, 16, 17].

4. OUR EXPERIMENT

4.1. Contribution I: compressing PINNs

We represented a solution $u = u(t, x)$ to 1D Burger’s equation by an NN [10]. In mathematical form, 1D Burger’s equation is

$$\begin{aligned} u_t + uu_x - (0.01/\pi)u_{xx} &= 0, \quad t \in [0, 1], \\ u(0, x) &= -\sin(\pi x), \\ u(t, -1) &= u(t, 1) = 0. \end{aligned} \quad (3)$$

When we used the NN with 8 hidden layers, and each layer comprises 100 neurons, its trainable parameters are amounted

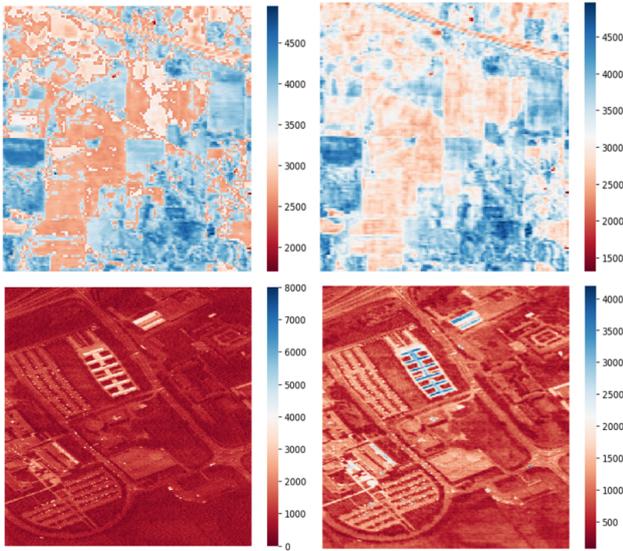


Fig. 4. Two examples of HSIs before and after tensor decomposition: [Top] The band 74 of the Indian Pine HSI before and after tensor decomposition, and [Bottom] The band 1 of the Pavia University HSI before and after tensor decomposition.

to 71, 101 parameters. We reduced these 71, 101 parameters to 32, 701 parameters by compressing the odd numbers of the hidden layers by utilizing the TT-decomposition (see Fig. 2 [Bottom]) [7]. We found a solution u to the Burger's equation while utilizing both the original and compressed NNs. Furthermore, the compressed PINN generated a solution to the Burger's equation with high accuracy such as having been produced by its original PINN, while it occupies a smaller parameter space than its original one (see Fig. 3). More importantly, the compressed NNs can be also utilized to analyse and classify any real-world datasets as shown in the article [7, 18].

4.2. Contribution II: decomposing real-world data tensors in factor matrices

We decompose two practical HSIs shown in Fig. 1 in a very small number of factor matrices by using CP-decomposition expressed by Eq. (1) to improve their spectral resolution; we illustrate our method for decomposing these HSIs in Fig. 2 [Top]. In our experiment, we set the rank R of the CP-decomposition at 145. For the Indian Pine HSI, the decomposition time was 0.1711 seconds, the compression ratio was 60, and the R-squared value between the raw and the decomposed Indian Pine HSI was 0.9959. For the Pavia University HSI, the decomposition time was 1.1013 seconds, the compression ratio was 140, and the R-squared value between the raw and the decomposed Pavia University HSI was 0.9450.

From these results, we gained the insight that the HSIs can be stored efficiently in conventional storage devices when applying tensor decomposition to the practical HSIs. More importantly, we improved the spectral resolution of the HSIs. We presented some visual examples of our finding in Fig. 4.

5. CONCLUSION

This paper focused on designing and applying a quantum-inspired tensor-network to DL models and real-world data tensors. Our contribution is twofold: (I) We reduced the parameters of a DL model when compressing them by using TT-decomposition. As a DL model, we utilized a physics-informed neural network for finding a solution to 1D Burger's equation. The compressed model generates solutions to 1D Burger's equation with high accuracy such as having produced by its original one. (II) We improved the spectral resolution of hyperspectral images (i.e. data tensors) by decomposing them in sparse factor matrices through CP-decomposition. The decomposed data tensors are represented by sparse tensors, while the decomposition time was extremely small (around 1 second). Additionally, we can store these decomposed images (i.e. sparse tensors) efficiently and securely in distributed storage devices thanks to their sparse factor matrices. As practical HSIs, we used HSIs of Indian Pine, USA and of Pavia University, Italy.

As a future and on-going work, we invent and design quantum-inspired machine learning models for data-driven and model-driven practical problems. In addition, we invent and analyse DL models supported by quantum tensor networks [15, 16, 11, 17].

6. REFERENCES

- [1] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.
- [2] Kamal Choudhary, Brian DeCost, Chi Chen, Anubhav Jain, Francesca Tavazza, Ryan Cohn, Cheol Woo Park, Alok Choudhary, Ankit Agrawal, Simon J. L. Billinge, Elizabeth Holm, Shyue Ping Ong, and Chris Wolverton, "Recent advances and applications of deep learning methods in materials science," *npj Computational Materials*, vol. 8, no. 1, pp. 59, Apr 2022.
- [3] G. Cheng, X. Xie, J. Han, L. Guo, and G. S. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 3735–3756, 2020.
- [4] Dimitrios Marmanis, Mihai Datcu, Thomas Esch, and Uwe Stilla, "Deep learning earth observation classifica-

- tion using imagenet pretrained networks,” *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 1, pp. 105–109, 2016.
- [5] David Patterson, Joseph Gonzalez, Quoc Le, Chen Liang, Lluís-Miquel Munguia, Daniel Rothchild, David So, Maud Texier, and Jeff Dean, “Carbon emissions and large neural network training,” 2021.
- [6] Andrzej Cichocki, “Era of big data processing: A new approach via tensor networks and tensor decompositions,” *CoRR*, vol. abs/1403.2048, 2014.
- [7] Ze-Feng Gao, Song Cheng, Rong-Qiang He, Z. Y. Xie, Hui-Hai Zhao, Zhong-Yi Lu, and Tao Xiang, “Compressing deep neural networks by matrix product operators,” *Phys. Rev. Res.*, vol. 2, pp. 023300, Jun 2020.
- [8] Jiaqi Gu, Ben Keller, Jean Kossaifi, Anima Anandkumar, Brucek Khailany, and David Z. Pan, “Heat: Hardware-efficient automatic tensor decomposition for transformer compression,” 2022.
- [9] Hao Huang, Xiao-Yang Liu, Weiqin Tong, Tao Zhang, Anwar Walid, and Xiaodong Wang, “High performance hierarchical tucker tensor learning using gpu tensor cores,” *IEEE Transactions on Computers*, pp. 1–1, 2022.
- [10] M. Raissi, P. Perdikaris, and G.E. Karniadakis, “Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations,” *Journal of Computational Physics*, vol. 378, pp. 686–707, 2019.
- [11] George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang, “Physics-informed machine learning,” *Nature Reviews Physics*, vol. 3, no. 6, pp. 422–440, Jun 2021.
- [12] Markus Reichstein, Gustau Camps-Valls, Bjorn Stevens, Martin Jung, Joachim Denzler, Nuno Carvalhais, and Prabhat, “Deep learning and process understanding for data-driven earth system science,” *Nature*, vol. 566, no. 7743, pp. 195–204, Feb 2019.
- [13] Nikita Gourianov, Michael Lubasch, Sergey Dolgov, Quincy Y. van den Berg, Hessam Babaei, Peyman Givi, Martin Kiffner, and Dieter Jaksch, “A quantum-inspired approach to exploit turbulence structures,” *Nature Computational Science*, vol. 2, no. 1, pp. 30–37, Jan 2022.
- [14] Chase Roberts, Ashley Milsted, Martin Ganahl, Adam Zalcman, Bruce Fontaine, Yijian Zou, Jack Hidary, Guifre Vidal, and Stefan Leichenauer, “TensorNetwork: A Library for Physics and Machine Learning,” 2019.
- [15] E Miles Stoudenmire, “Learning relevant features of data with multi-scale tensor networks,” *Quantum Science and Technology*, vol. 3, no. 3, pp. 034003, apr 2018.
- [16] William Huggins, Piyush Patil, Bradley Mitchell, K Birgitta Whaley, and E Miles Stoudenmire, “Towards quantum machine learning with tensor networks,” *Quantum Science and Technology*, vol. 4, no. 2, pp. 024001, jan 2019.
- [17] Ivan Glasser, Nicola Pancotti, and J. Ignacio Cirac, “From probabilistic graphical models to generalized tensor networks for supervised learning,” *IEEE Access*, vol. 8, pp. 68169–68182, 2020.
- [18] Xun Gao, Eric R. Anschuetz, Sheng-Tao Wang, J. Ignacio Cirac, and Mikhail D. Lukin, “Enhancing generative models via quantum correlations,” *Phys. Rev. X*, vol. 12, pp. 021037, May 2022.

5.1.2 Publication H: Exploiting the Quantum Advantage for Satellite Image Processing: Review and Assessment

S. Otgonbaatar and D. Kranzlmüller

Abstract. We discuss the current state of quantum computing in satellite imagery, and analyze the potential limitations and applications of QML models when dealing with satellite data. Additionally, we consider the ongoing challenges to profit from quantum advantage and to find the ideal balance between HPC and QC. We then assess some parameterized quantum circuit models transpiled into a Clifford+T universal gate set. The T-gates reveal the quantum resources needed to deploy quantum models on an HPC system or several QC systems. Specifically, if T-gates cannot be simulated efficiently on an HPC system, we can apply a quantum computer and its computational power instead. Our quantum resource estimation shows that QML models, with a sufficient number of T-gates, provide the quantum advantage only if they generalize on unseen data points better than their classical counterparts deployed on the HPC system, and if they break the symmetry in their weights at each learning iteration as in conventional deep neural networks. We also estimate the quantum resources required for some QML models as an initial innovation. Lastly, we define the optimal sharing between an HPC+QC system for executing QML models for hyperspectral images. These images are a unique dataset compared with other satellite images since they have a limited number of input qubits and a small number of labeled benchmark images, making it less challenging to deploy them on a small-scale quantum computer.

Received 5 September 2023; revised 13 November 2023; accepted 28 November 2023; date of publication 4 December 2023; date of current version 2 January 2024.

Digital Object Identifier 10.1109/TQE.2023.3338970

Exploiting the Quantum Advantage for Satellite Image Processing: Review and Assessment

SORONZONBOLD OTGONBAATAR^{1,2}  AND DIETER KRANZLMÜLLER² 

¹Remote Sensing Technology Institute, German Aerospace Center DLR, 82234 Weßling, Germany

²Institut für Informatik, Ludwig-Maximilians-Universität München, 80538 München, Germany

Corresponding author: Soronzonbold Otgonbaatar (e-mail: soronzonbold.otgonbaatar@dlr.de).

This work was supported by the European Space Agency through Project Quantum Advantage for Earth Observation (QA4EO).

ABSTRACT This article examines the current status of quantum computing (QC) in Earth observation and satellite imagery. We analyze the potential limitations and applications of quantum learning models when dealing with satellite data, considering the persistent challenges of profiting from quantum advantage and finding the optimal sharing between high-performance computing (HPC) and QC. We then assess some parameterized quantum circuit models transpiled into a Clifford+T universal gate set. The T-gates shed light on the quantum resources required to deploy quantum models, either on an HPC system or several QC systems. In particular, if the T-gates cannot be simulated efficiently on an HPC system, we can apply a quantum computer and its computational power over conventional techniques. Our quantum resource estimation showed that quantum machine learning (QML) models, with a sufficient number of T-gates, provide the quantum advantage if and only if they generalize on unseen data points better than their classical counterparts deployed on the HPC system and they break the symmetry in their weights at each learning iteration like in conventional deep neural networks. We also estimated the quantum resources required for some QML models as an initial innovation. Lastly, we defined the optimal sharing between an HPC+QC system for executing QML models for hyperspectral satellite images. These are a unique dataset compared with other satellite images since they have a limited number of input quantum bits and a small number of labeled benchmark images, making them less challenging to deploy on quantum computers.

INDEX TERMS Earth observation (EO), hyperspectral images, image classification, quantum computers, quantum machine learning (QML), quantum resource estimation, remote sensing.

I. INTRODUCTION

A. WHY QUANTUM COMPUTING FOR EARTH OBSERVATION?

Earth observation (EO) methodologies tackle optimization and artificial intelligence (AI) problems involving big datasets obtained from instruments mounted on spaceborne and airborne platforms. Some optimization and AI problems combined with big EO datasets are intractable computational problems for conventional high-performance computing (HPC) systems. In addition, EO datasets themselves are complex heterogeneous image datasets, compared with conventional red-green-blue images, characterized by so-called 4 V features comprising *volume*, *variety*, *velocity*, and *veracity* [1]; here, *volume* refers to big EO datasets (e.g., terabytes of data per day collected, for instance, by the European Space Agency); *variety* refers to distinct spectral data, such

as multispectral and hyperspectral pixel data; *velocity* refers to the speed of change on the Earth's surface; and *veracity* refers to imperfect datasets, such as noisy images or remotely sensed images, partly covered by clouds [2]. In general, EO problems also include calibration and integer optimization problems in synthetic aperture radar applications [3], [4], a Bayesian paradigm (e.g., Gaussian processes) for retrieving physical parameters from remotely sensed datasets [5], [6], uncertainty estimates for EO predictions [7], solving partial differential equations (PDEs) for climate modeling and digital twin Earth paradigms [8], and identifying objects on the Earth's surface [9]. Furthermore, some computational problems like AI training architectures are computationally expensive and inherently intractable problems or NP-hard problems (see Fig. 1) [10]; nondeterministic (NP) polynomial problems are computational problems where there

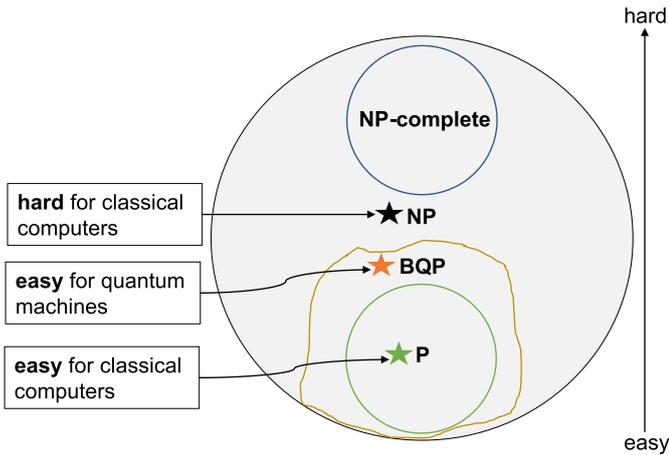


FIGURE 1. Computational complexity conjecture draws boundaries between computational problems according to their hardness based on the required classical and quantum computational resources. In particular, the computational problem denoted by the green star is easy to solve for both quantum machines and classical computers, the computational problem denoted by the orange star is easy for quantum machines but hard for classical machines, and the computational problem denoted by the black star is hard for classical computers. Still, no known efficient quantum algorithmic approaches exist for quantum machines.

are no known efficient commonly used algorithms for finding their solutions in a reasonable polynomial time (i.e., a polynomial number of steps) but can be verified in a polynomial time given their solutions, and NP-hard problems are computational problems harder than NP problems. On the other hand, quantum machines harnessing quantum physics phenomena like entanglement can solve some challenging problems faster and more efficiently than their counterpart conventional machines ranging from integer optimization problems [11], [12], [13] to AI techniques [14], [15], [16], [17], [18], and PDEs, [19], [20], and even quantum-inspired algorithms for solving PDEs [21]. Thus, quantum algorithms’ computational advantages (or quantum advantage) over conventional algorithms inspire enough to examine and identify computationally intractable problems with EO methodologies and hard EO datasets for near- and far-term quantum machines. We note that the terms “quantum machine” and “quantum computer” are generally interchangeable. However, the former is used to describe current quantum platforms that operate at a hardware level, rather than at the level of a classical computer.

B. DO WE REALLY NEED QUANTUM MACHINES?

Quantum machines can be generally divided into three families comprising quantum annealers [22], quantum simulators [23], [24], and universal quantum computers [25]. These quantum machines promise computational advantage for computing notoriously difficult problems over conventional computers according to computational complexity theorems/conjectures [26], [27]; computational complexity theorems draw boundaries between computational problems according to their hardness for finding their solutions (see Fig. 1) [10]. At the moment, quantum machines are designed

to tackle specific forms and kinds of intractable computational problems, e.g., quantum annealers for quadratic unconstrained binary optimization (QUBO) problems or simulating the Ising Hamiltonian [11], and quantum simulators for mimicking some physical Hamiltonian [28], [29]. Research communities ranging from high-energy physics [24], condensed-matter physics [29], AI [15], to EO [30] are in the exploration phase of identifying and investigating their hard problems for quantum platforms. Furthermore, classical computational methods for intractable computational problems reach their limitations and potential accuracy due to the classical computational resource required and the complexity of both EO challenges and datasets. As stated earlier, some computational techniques are intractable problems on conventional machines and computationally expensive, even on the HPC system. Thus, to go beyond current computational methods integrated with large-scale datasets to find a better solution and utilize low computational cost, it is inevitable to examine and identify computationally demanding problems in EO applications for novel near- and long-term quantum machines. More importantly, gaining insight into programming these novel computing machines and their potential advantages and imperfections for computational problems is vital.

C. STATE OF THE ART OF QC FOR EO

Quantum computing (QC) is a novel computing paradigm that promises to find solutions to some intractable computational problems more efficiently and faster by exploiting quantum superposition and entanglement than conventional computing techniques if and only if one considers ideal quantum complexity measures without overhead considerations like a distillation of Toffoli gates in the real quantum machines, e.g., the classical versions of the Toffoli gates are transistors in a conventional computer [31]. Quantum machines are a kind of computer constructed using the primitives of a QC method, such as quantum bits (qubits) and quantum gates, in contrast to traditional classical bits and transistors. Digital quantum machines can be decomposed into the following three layers [32]:

- 1) a *quantum state preparation* or a *quantum data encoding* layer;
- 2) a *quantum unitary evolution* or a *parametrized quantum gate* layer;
- 3) a *quantum measurement* layer.

For gaining insight into computing EO problems involving big datasets on quantum machines, some studies already exist for processing a *variety* of EO datasets to tackle EO challenges using hybrid classical–quantum approaches (see Fig. 2). Hybrid classical–quantum approaches involve the use of a classical computer to enhance quantum algorithms. Quantum machine learning (QML) is a type of hybrid classical–quantum approach, which is interchangeable with quantum artificial intelligence. It is also worth noting that a

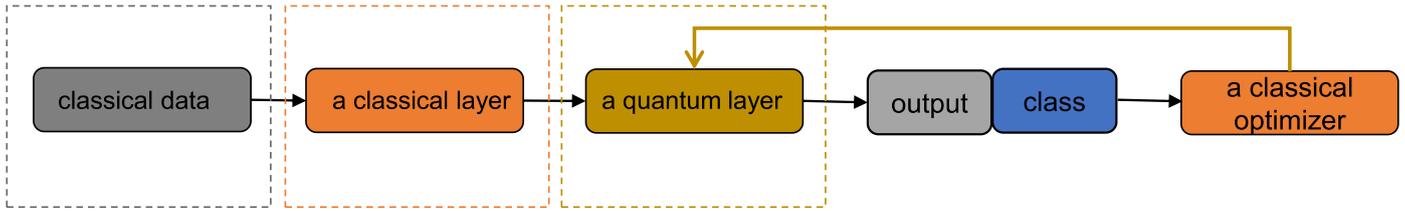


FIGURE 2. Hybrid classical-quantum approach for computational and machine learning tasks. A quantum layer includes implicitly quantum data encoding, parametrized quantum gates, and quantum measurement layers.

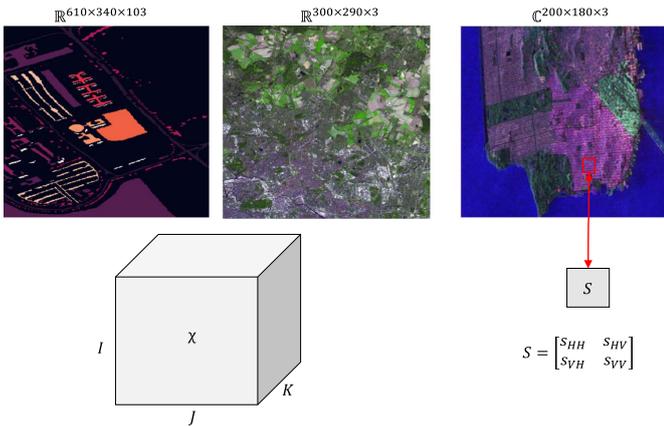


FIGURE 3. (Top) Example hyperspectral, multispectral, and polarimetric images, (bottom Left) their third-order tensor representation, and (bottom right) each pixel/target in polarimetric images is characterized by the complex numbered scattering matrix in contrast to hyperspectral and multispectral images. Here, s_{ij} denotes a scattering element given sent/reflected horizontal H or vertical V polarized beam.

variety of datasets includes hyperspectral, multispectral, and polarimetric EO images.

1) EO IMAGES

We can generalize that EO images are third-order tensors regardless of a *variety*. Furthermore, a hyperspectral image is a remotely sensed image denoted by $\mathbb{R}^{I \times J \times K}$, where I and J are its spatial dimensionality, and K means hundreds of its narrow-spaced spectral bands (or features), e.g., the Pavia University, hyperspectral image described by $\mathbb{R}^{610 \times 340 \times 103}$ tensor. Multispectral images are a third-order tensor $\mathbb{R}^{I \times J \times K}$ with at most $K = 12$ spectral bands. The main difference between them is the spectral bands' number and spacing. In contrast, polarimetric images are characterized by the scattering property S of ground targets; each pixel is described by a 2×2 scattering matrix but not by spectral bands as in hyperspectral and multispectral images. Hence, we could assume that polarimetric images have $K = 3$ informative features if the scattering matrix is symmetric and $K = 4$ informative features otherwise (see Fig. 3) [33].

2) QML FOR EO IMAGES

Climate AI tasks involve analyzing satellite images that consist of thousands of pixels and hundreds of spectral bands. For example, Eurosat multispectral images have a size of

64×64 pixels and 12 spectral bands, which can be represented as $\mathbb{R}^{64 \times 64 \times 12}$ [34]. In contrast, the digital quantum machines currently available on the market have less than a hundred noisy qubits and around depth-five of faulty quantum gates [35]. Therefore, the main challenge is to embed satellite images in a *quantum data encoding* layer, regardless of the size of quantum machines and their quantum errors. To address this challenge, the authors in [36] and [37] proposed and utilized a two-level embedding scheme. This scheme comprises a classical layer for dimensionality reduction and a *quantum data encoding* layer for dimensionally reduced images. In other words, they used a hybrid classical-quantum approach, embedding classical datasets in a *quantum data encoding* layer and optimizing a *parametrized quantum gate* layer of digital quantum computers with the help of a conventional classical computer. However, the Eurosat dataset they used is a large dataset consisting of low-dimensional and easy-to-classify images and thus has low *veracity*. Most EO datasets, on the other hand, are small datasets containing high-dimensional and hard-to-classify images or high *veracity* images. For example, the multispectral UC Merced Land Use dataset has a size of 245×245 pixels and three spectral bands, which can be represented as $\mathbb{R}^{245 \times 245 \times 3}$ [38]. To investigate the performance of quantum machines with varying depths of a *parametrized quantum gate* layer, Otgonbaatar et al. [39] utilized this dataset and polarimetric EO images for natural embedding in input qubits without a dimensionality reduction technique [40]. It is important to note that the quality of the datasets used plays a crucial role in data-driven tasks for hybrid classical-quantum approaches [41]. Therefore, Gupta et al. [42] analyzed the power of EO image datasets for training digital quantum machines.

Furthermore, a quantum annealer is a type of quantum simulator that is designed to simulate an Ising Hamiltonian equivalent to QUBO problems [22]. The authors in [43], [44] analyzed classification problems posed as QUBO problems, belonging to NP-hard problems, on a D-Wave quantum annealer. They employed binary hyperspectral EO images since a D-Wave quantum annealer promises to converge to a better solution to NP-hard problems. Some studies also transformed a support vector machine (SVM) into a QUBO problem [45] and optimized it on a D-Wave quantum annealer when analyzing EO image datasets [33], [46]. To embed large EO datasets in a D-Wave quantum annealer, Otgonbaatar et al. [47] used a K-fold technique and the concept of a coreset since a D-Wave quantum annealer has around

5000 qubits arranged according to an expressly limited topology. A D-Wave quantum annealer was also proposed for a notoriously hard feature selection task and multilabel SVM for remotely sensed hyperspectral images [48].

Lastly, quantum-inspired algorithms are becoming increasingly popular in both academic and industrial circles due to their energy and computational efficiency. These algorithms are inspired by the potential advantages of quantum algorithms, such as the quantum-inspired quantum Fourier transformation [49], quantum-inspired AI/ML [50], and the use of tensor networks to compress deep neural networks [51]. Tensor networks are designed to compute quantum many-body systems efficiently [52], and they are currently being used to simulate quantum circuits on modern GPU tensor cores [53]. Thanks to these advancements, quantum tensor networks have been successfully utilized to decrease the weights of physics-informed neural networks and increase the resolution of hyperspectral images [54].

3) SELECTING EO DATA FOR QUANTUM MACHINES

When working with quantum machines in EO challenges, it is vital to choose remotely sensed datasets based on the principle that “the more features in the dataset, the less quantum resources required.” Studies have shown that processing multispectral images requires more quantum gates and qubits than hyperspectral and polarimetric images [36], [40]. This is because multispectral images need global feature capturing, with each pixel dependent on its neighbors, making processing more resource-intensive. On the other hand, hyperspectral and polarimetric images contain informative spectral information for each pixel. They can be embedded in qubits without the constraint of their neighbors, making processing less resource-intensive [40]. For instance, one QML model known as a quantum convolutional neural network (QCNN) requires approximately 4000 quantum gates only to embed the element $\mathbb{R}^{64 \times 64 \times 12}$ in the Eurosat dataset and roughly 60 000 quantum gates for embedding the multispectral image $\mathbb{R}^{300 \times 290 \times 3}$ illustrated in Fig. 3 in the input qubits [55]. Hence, multispectral images are not viable for deploying QCNNs on today’s quantum machines, even on future quantum machines. However, a hybrid classical-quantum model requires fewer quantum resources than QCNNs. Otgonbaatar and Datcu [36] used only 16 quantum gates for encoding the Eurosat and the multispectral image $\mathbb{R}^{300 \times 290 \times 3}$ depending on the compressing quality. In contrast, we can embed the pixels of a hyperspectral image, e.g., the Pavia University hyperspectral image, in the input qubits using only at least three and, at most, about 103 quantum gates thanks to their abundant spectral bands [33]. As for polarimetric images, we need at most five quantum gates due to their doppelgänger feature to qubits or the one-to-one mapping between polarimetric images and qubits [40].

Based on the above analysis, hyperspectral satellite images are much more appropriate for designing and assessing QML models and tackling climate challenges than multispectral and polarimetric images since they have abundant spectral

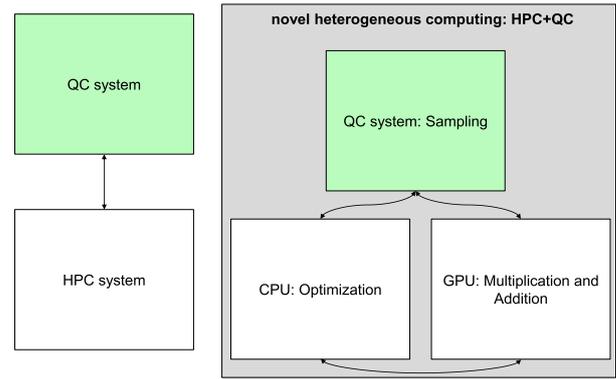


FIGURE 4. Novel heterogeneous computing. A high-performance and QC paradigm. Here, conventional heterogeneous computing refers to the programming of CPU and GPU, whereas we call novel heterogeneous computing when integrating QPUs with CPUs and GPUs. QPUs can be several parallel quantum machines based on different quantum technologies such as quantum annealing, neutral atoms, superconducting, and photonic.

information and fewer quantum resources required than other remotely sensed datasets. More importantly, QML models generalize better on small-scale datasets than their classical alternatives [56], whereas a hyperspectral dataset has limited labeled images (or small-scale datasets) compared to multispectral datasets and has more features than both multispectral and polarimetric datasets.

D. HOW AND WHEN DO QUANTUM MACHINES OUTPERFORM CONVENTIONAL COMPUTERS?

It is becoming increasingly clear that quantum processing units (QPUs) will soon be working alongside conventional classical computers, such as how central processing units (CPUs) and graphics processing units (GPUs), are used in heterogeneous computing [30]. We are currently in the era of HPC, and the emergence of QC is a new and exciting concept in heterogeneous computing. It involves integrating a CPU+GPU with QPUs designed to handle specific computational problems (see Fig. 4). For instance, a quantum annealer is designed to tackle only QUBO problems, and neutral atom platforms can simulate certain chemical Hamiltonians. Depending on the difficulty level of the computational problems, we may need to program a challenging heterogeneous computing environment (i.e., CPU+GPU+QPUs) or a conventional one (i.e., CPU+GPU).

QPUs, except for quantum annealers, currently consist of around 100 error-prone qubits and low-depth, faulty quantum gates. Preskill [57] coined these devices as “noisy intermediate-scale quantum (NISQ) devices.” However, for practical computational problems, there is no demonstration of the computational advantage of NISQ devices over a conventional classical computer. Therefore, estimating the quantum resources required to tackle hard computational and ML problems is vital to achieving a quantum advantage in EO. It is worth noting that some quantum algorithms can be simulated efficiently using a conventional classical computer. For

this reason, any reasonable quantum resource estimation of a quantum algorithm should consider non-Clifford T-gates, error rates of qubits and quantum gates, and the execution time of single- and two-qubit quantum gates [58].

Non-Clifford T-gates are the most resource-expensive part of implementing a quantum algorithm, compared with Clifford quantum gates or CNOT, Hadamard, Phase, and measurement gates. Even the Gottesman–Knill theorem states (informally) that non-Clifford T-gates cannot be efficiently simulated on a conventional classical computer. In contrast, Clifford quantum gates can be simulated in polynomial time using a conventional classical computer without any restriction on entanglement [58], [59]. Specifically, quantum algorithms consisting only of Clifford quantum gates can be simulated in $\mathcal{O}(n^2 \sim m)$ polynomial steps with n qubits and m Clifford operations. However, quantum algorithms consisting of Clifford+T gates take exponential steps $\mathcal{O}(\kappa t^3 \epsilon^{-2})$, with the number of T-gates known as T count (t), stabilizer state (k) growing exponentially $\mathcal{O}(2^t)$, and an error rate (ϵ) [58]. We note that some quantum algorithms can be efficiently simulated using a sophisticated classical technique like a tensor network on GPU tensor cores [60].

The Clifford+T gate set $\{S, H, \text{CNOT}, T\}$ is considered a universal gate set for digital QPUs. This is due to the feasibility of quantum error-correcting, known as a surface code [61]. More importantly, the surface code enables the creation of fault-tolerant digital quantum computers that surpass the NISQ-era computers [35]. In contrast to NISQ computers, fault-tolerant quantum computers are made up of error-free qubits and quantum gates that are transpiled into the Clifford+T gate set. Therefore, this shows that for quantum advantage in EO applications to be reached if and only if our quantum learning models have a sufficiently high number $\mathcal{O}(10^{12})$ of T-gates and generalize on unseen data points [62]. Otherwise, we can simulate them efficiently using conventional classical computing resources.

Further, a hybrid classical-quantum approach for computational EO problems is embedding high-dimensional classical data in a limited number of qubits and optimizing the weights of a parameterized quantum model [36], [63]. There is yet another challenging question: how notoriously difficult computational problems can take advantage of both HPC and QC systems or when we should execute them on an HPC instead of a QC system and vice versa. We decompose the parameterized quantum model into the Clifford+T gate set at each learning iteration to tackle these issues. If the parameterized quantum model only includes Clifford gates and a small number of T-gates [64], then we execute it on the HPC system instead of the QC machines since we already know that Clifford gates and hundreds of T-gates can be simulated efficiently using a conventional classical computer. We re-emphasize that quantum learning models can be simulated efficiently using a classical computer without the need for quantum computers if they do not have a high number of T-gates.

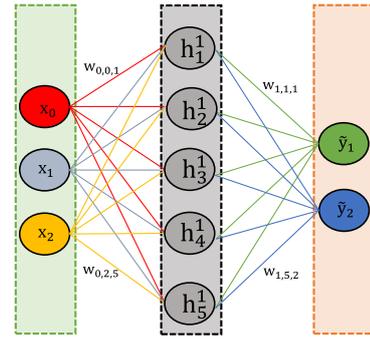


FIGURE 5. Visual representation of traditional neural networks.

1) QML–SYMMETRY-BREAKING

Symmetry-breaking refers to asymmetric tunable weights of traditional ML models such that the weights capture and rank the dataset’s features during training. Consider a neural network with a single hidden layer illustrated in Fig. 5. Mathematically, it is defined

$$h_i^1 = f \left(w_{0,i} + \sum_{j=0}^2 w_{0,j,i} x_j \right), \quad i = 1, \dots, 5 \quad (1)$$

$$\tilde{y}_l = f \left(h_{0,l} + \sum_{j=1}^5 w_{1,j,l} h_j^1 \right), \quad l = 1, 2 \quad (2)$$

where $f(\cdot)$ is a nonlinear activation function, w ’s denotes a tuneable weight, and x_k is the dataset’s feature. We note that w ’s must have different values identical to a linear regression model $\tilde{y} \sim w_0 + w_1 \cdot x_0 + w_2 \cdot x_1$. If the model weights are symmetric $w_1 = w_2$, it has not learned the dataset’s feature. To capture the dataset’s feature, the learning model must have asymmetric weights $w_1 \neq w_2$, or the learning model must break the symmetry in its weights. Identical to the symmetry-breaking in conventional ML, Haug et al. [18] implicitly demonstrated that QML models also must break symmetry in their weights, resulting in better generalizability or more expressive power and higher effective dimension than their classical counterparts. In particular, they identified and disregarded some redundant weights in their quantum models that are symmetric (e.g., the same digital values) and do not simultaneously increase the QML model’s expressive power. They, however, did not estimate the hardness of their QML models characterized by non-Clifford T-gates that can be implemented efficiently on quantum machines and otherwise difficult on conventional HPC systems.

Furthermore, to outperform classical learning models deployed on an HPC system, we should invent and design QML models having thousands of T-gates, and their expressive power (signaling the symmetry-breaking in QML models) is higher than their classical counterparts [16]. There is (still) no such QML model with thousands of T-gates and higher expressive power on unseen data points than its classical counterpart.

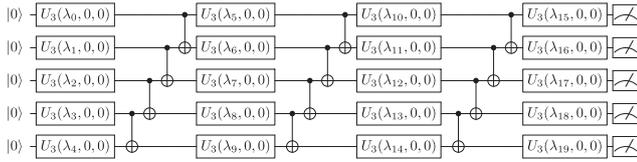


FIGURE 6. We transpiled a real-amplitude quantum circuit having depth-one into the Clifford+T and the native gate set. It is used to demonstrate the power of a PQC model by Abbas et al. [16].

II. QUANTUM RESOURCE ESTIMATION FOR HYPERSPECTRAL IMAGES

A hyperspectral imaging satellite, such as the EnMAP satellite,¹ is a type of imaging instrument mounted on a satellite and used to sense spectral reflectances. The mission of this satellite is to collect hyperspectral imaging data that provides crucial information for scientific inquiries, societal grand challenges, and key stakeholders and decision-makers. This information pertains to various topics, such as climate change impact and interventions, hazard and risk assessment, biodiversity and ecosystem processes, land cover changes, and surface processes.

We already have seen that hyperspectral images require less quantum resources than other remotely sensed datasets. They also have limited label information, and there is limited availability of benchmark hyperspectral images compared with conventional benchmark remote-sensing datasets, such as multispectral images [65], [66]. When training QML models on limited benchmark-oriented labeled hyperspectral image datasets, a classical layer can reduce the dimensionality of the hyperspectral image dataset's spectral bands due to the limited number of input qubits. However, the degree of dimensionality reduction required for the given hyperspectral image dataset depends on the utilized quantum machines. Regardless of their error, this means whether we can access a quantum machine with qubits ≤ 100 or > 100 . The role of classical machines in preprocessing the hyperspectral image dataset is reduced as we can feed many informative features to a quantum machine with less dimensionality reduction, especially as the number of qubits of quantum machines increases. We assume we used EnMAP hyperspectral images with 103 spectral bands and 610×340 spatial dimensions. The EnMAP hyperspectral images also have 207 400 data points and 103 features, which are small-scale image datasets compared with conventional multispectral images. To execute the QML model on the quantum machine having ≤ 100 input qubits, we can either reduce the spectral bands of the EnMAP hyperspectral images from 103 to at most 100 or select the most informative 100 bands to be compatible with the input qubits by utilizing a classical machine. Instead, for quantum machines with more than 100 input qubits, we can use a classical machine to persevere more spectral bands of the EnMAP hyperspectral images when performing the dimensionality reduction or the feature selection technique in the spectral bands.

¹<https://www.enmap.org/mission/>

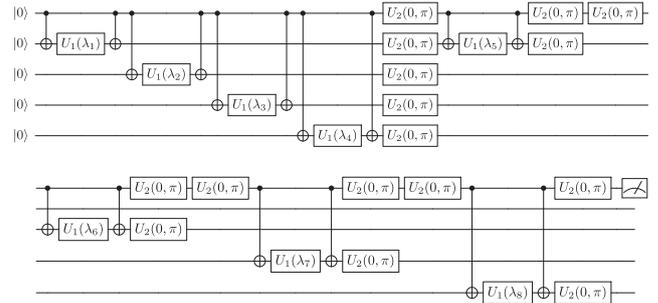


FIGURE 7. We transpiled an energy-based quantum circuit having depth-one into the Clifford+T and the native gate set. This PQC model is proposed for the NISQ device by Farhi and Neven [68].

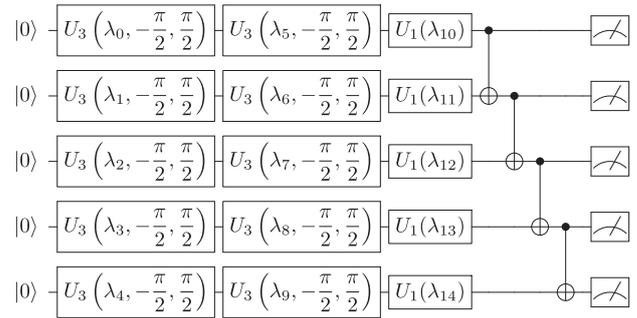


FIGURE 8. We transpiled a strongly entangling quantum circuit having depth-one transpiled into the Clifford+T and the native gate set. This PQC model is proposed to build a powerful quantum learning model by Schuld et al. [73].

Toward quantum resource estimation, we assessed four different PQC models expressed by the Clifford+T gate set (see Figs. 6–9). The Clifford+T gate set is defined by U_1 , U_2 , U_3 , and CNOT gates

$$U_1(\lambda) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{pmatrix} \quad U_2(\lambda, \phi) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & -e^{i\phi} \\ e^{i\lambda} & e^{i(\lambda+\phi)} \end{pmatrix}$$

$$U_3(\lambda, \phi, \gamma) = \begin{pmatrix} \cos(\gamma/2) & -e^{i\lambda} \sin(\gamma/2) \\ -e^{i\phi} \sin(\gamma/2) & e^{i(\phi+\lambda)} \cos(\gamma/2) \end{pmatrix} \quad (3)$$

where for example, $U_1(\pi/4) = U_3(\pi/4, 0, 0) = T$, $U_1(\pi/2) = S$, $U_2(0, \pi) = H$. Hence, the Clifford+T gate set can be $\{U_1(\pi/2), U_2(0, \pi), \text{CNOT}, U_1(\pi/4)\}$, and a hardware-specific native gate set is $\{U_1(\lambda), U_2(\lambda, \phi), U_3(\lambda, \phi, \gamma), \text{CNOT}\}$.

We have chosen the PQC models in Figs. 6–9 as benchmark QML models identical to conventional benchmark deep learning (DL) models, such as Resnet [67]. The quantum resource required for executing them on the quantum machine is $\mathcal{O}(1)$ (constant time) if there is either no sign of T-gates or a low number of T-gates. In particular, we will deploy them on either the HPC system or the quantum machines depending on the existence and the number of T-gates in their configuration during the training phase. Furthermore, the number of T-gates defines the quantum resource required for deploying QML models on quantum computers.

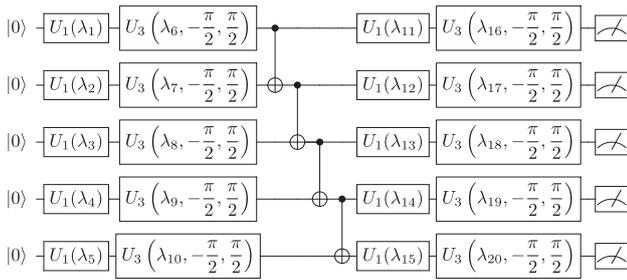


FIGURE 9. We transpiled a hardware-efficient quantum circuit having depth-one into the Clifford+T and the native gate set. This PQC is used for quantum variational inference by Benedetti et al. [74].

We used the symmetry-breaking concept inherited from conventional neural networks to determine the number of T-gates in our four PQCs [69]. Again, we strongly emphasize that QML models break the symmetry in their weights to decrease their redundant parameterized quantum gates, resulting in better generalization on unseen data points than conventional neural networks [18]. Namely, each weight within a parameterized quantum layer must have different digital values for capturing unique features. Therefore, we assumed that each layer of the QML models must have, at most, a single T-gate at each learning iteration, and our QML models having depth-one can only have one T-gate.

As for the quantum resource required for executing them on the quantum hardware, we assumed also the following.

- 1) If our PQCs have 10^8 T-gates and five logical qubits then we need 158 431 physical qubits (i.e., 9375 state distillation qubits, and 149 056 physical qubits) with a surface code distance of $d = 25$, and our QML models then take around 5 h per shot.
- 2) If our PQCs have three T-gates and five logical qubits then we need 50 700 physical qubits (i.e., 14 400 state distillation qubits, and 36 300 physical qubits) with a surface code distance of $d = 11$, and our QML models then take around $8.12 \cdot 10^{-8}$ h per shot.
- 3) If our PQCs have one T-gate and five logical qubits, then we need 15 135 physical qubits (i.e. 14 400 state distillation qubits, and 735 physical qubits) with a surface code distance of $d = 7$, and our QML models then take around $2.07 \cdot 10^{-8}$ h per shot.

Based on the study of the authors in [70] and [71], we estimated the quantum resources required for deploying QML models on error-correcting quantum machines known as surface code quantum computers. Our estimation considers that the quantum gate error is about $p = 10^{-3}$, and the single round of the surface code takes around 10^{-6} s. Here, the hours refer to T-gates preparation; Fowler and Gidney [70] provided a detailed spreadsheet for the quantum resource estimation. The quantum resource estimation demonstrates whether the QML models have to be deployed on quantum computers or not [64], [72], and it also generates the number of physical qubits required for deploying quantum algorithms on the surface code quantum computers.

III. CONCLUSION

We assessed the quantum resource required to execute QML models on a digital quantum computer to obtain a quantum advantage. We demonstrated that some quantum advantage can only be obtained if and only if QML models have a sufficient number of T-gates and generalize better on unseen data points than their classical counterparts. To count the T-gates of a particular QML model, we used the strong assumption that the QML models must break the symmetry in their weights—identical to the symmetry-breaking in conventional deep learning models—so that they become a more powerful model than their counterpart classical learning models. Based on the number of T-gates, we proposed a new HPC+QC paradigm (novel heterogeneous computing). In particular, we can simulate QML models on an HPC system (i.e., CPU+GPU) if they comprise a few hundred T-gates.

Toward quantum advantage in EO, we focused on QML models for hyperspectral images acquired by the EnMAP satellite since QML models can be trained on a limited labeled dataset, and our hyperspectral images have limited label information compared with multispectral images. For QML models, we utilized four parameterized quantum circuits and estimated the quantum resources required for deploying them on digital quantum machines. We found that we can deploy our QML models on an HPC system instead of a QC system since they only have a single T-gate due to the symmetry-breaking assumption. To design QML models with around $\mathcal{O}(10^8)$ that cannot be executed on an HPC system, they must have almost a depth of $\mathcal{O}(10^8)$, which is impractical for current and future quantum computers. Toward quantum advantage, it seems, therefore, reasonable to build, first, a special-purpose digital quantum computer for some practically significant computational task instead of a universal digital quantum computer similar to a D-Wave quantum annealer.

As future and ongoing work, we will invent and design a QML model with a reasonable depth that cannot be simulated on HPC systems but can be executed efficiently on QC systems and simultaneously has more expressive power over classical learning models.

ACKNOWLEDGMENT

The authors would like to thank the Gottfried Schwarz for reading, commenting, and improving the quality of this paper. This article is also part of the exploratory study dubbed “Quantum Advantage for Earth Observation (QA4EO)” granted by the European Space Agency.

REFERENCES

- [1] M. Reichstein et al., “Deep learning and process understanding for data-driven Earth system science,” *Nature*, vol. 566, no. 7743, pp. 195–204, Feb. 2019, doi: [10.1038/s41586-019-0912-1](https://doi.org/10.1038/s41586-019-0912-1).
- [2] P. Ebel, Y. Xu, M. Schmitt, and X. X. Zhu, “SEN12MS-CR-TS: A remote-sensing data set for multimodal multitemporal cloud removal,” *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 5222414, doi: [10.1109/TGRS.2022.3146246](https://doi.org/10.1109/TGRS.2022.3146246).

- [3] Y. Shi, X. X. Zhu, and R. Bamler, "Nonlocal compressive sensing-based SAR tomography," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 5, pp. 3015–3024, May 2019, doi: [10.1109/TGRS.2018.2879382](https://doi.org/10.1109/TGRS.2018.2879382).
- [4] X. X. Zhu et al., "Deep learning meets SAR: Concepts, models, pitfalls, and perspectives," *IEEE Geosci. Remote Sens. Mag.*, vol. 9, no. 4, pp. 143–172, Dec. 2021, doi: [10.1109/MGRS.2020.3046356](https://doi.org/10.1109/MGRS.2020.3046356).
- [5] G. Camps-Valls, J. Verrelst, J. Munoz-Mari, V. Laparra, F. Mateo-Jimenez, and J. Gomez-Dans, "A survey on Gaussian processes for Earth-observation data analysis: A comprehensive investigation," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 2, pp. 58–78, Jun. 2016, doi: [10.1109/MGRS.2015.2510084](https://doi.org/10.1109/MGRS.2015.2510084).
- [6] D. Narmandakh, C. Butscher, F. D. Ardejani, H. Yang, T. Nagel, and R. Taherdangkoo, "The use of feed-forward and cascade-forward neural networks to determine swelling potential of clayey soils," *Comput. Geotechnics*, vol. 157, 2023, Art. no. 105319, doi: [10.1016/j.compgeo.2023.105319](https://doi.org/10.1016/j.compgeo.2023.105319).
- [7] A. Stritih, P. Bebi, and A. Grêt-Regamey, "Quantifying uncertainties in Earth observation-based ecosystem service assessments," *Environ. Model. Softw.*, vol. 111, pp. 300–310, 2019, doi: [10.1016/j.envsoft.2018.09.005](https://doi.org/10.1016/j.envsoft.2018.09.005).
- [8] J. Pathak et al., "FourCastNet: A global data-driven high-resolution weather model using adaptive fourier neural operators," 2022, doi: [10.48550/arXiv.2202.11214](https://doi.org/10.48550/arXiv.2202.11214).
- [9] G. Cheng, X. Xie, J. Han, L. Guo, and G. S. Xia, "Remote sensing image scene classification meets deep learning: Challenges, methods, benchmarks, and opportunities," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 3735–3756, 2020, doi: [10.1109/JSTARS.2020.3005403](https://doi.org/10.1109/JSTARS.2020.3005403).
- [10] S. Arora and B. Barak, *Computational Complexity: A Modern Approach*. Cambridge, U.K.: Cambridge Univ. Press, 2009. [Online]. Available: <https://www.cambridge.org/us/catalogue/catalogue.asp?isbn=9780521424264>
- [11] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser, "Quantum computation by adiabatic evolution," 2000, doi: [10.48550/arXiv.quant-ph/0001106](https://doi.org/10.48550/arXiv.quant-ph/0001106).
- [12] A. Lucas, "Ising formulations of many NP problems," *Front. Phys.*, vol. 2, 2014, doi: [10.3389/fphy.2014.00005](https://doi.org/10.3389/fphy.2014.00005).
- [13] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for Big Data classification," *Phys. Rev. Lett.*, vol. 113, Sep. 2014, Art. no. 130503, doi: [10.1103/PhysRevLett.113.130503](https://doi.org/10.1103/PhysRevLett.113.130503).
- [14] J. Allcock, C.-Y. Hsieh, I. Kerenidis, and S. Zhang, "Quantum algorithms for feedforward neural networks," *ACM Trans. Quantum Comput.*, vol. 1, no. 1, pp. 1–24, Oct. 2020, doi: [10.1145/3411466](https://doi.org/10.1145/3411466).
- [15] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, no. 7671, pp. 195–202, Sep. 2017, doi: [10.1038/nature23474](https://doi.org/10.1038/nature23474).
- [16] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, "The power of quantum neural networks," *Nature Comput. Sci.*, vol. 1, no. 6, pp. 403–409, Jun. 2021, doi: [10.1038/s43588-021-00084-1](https://doi.org/10.1038/s43588-021-00084-1).
- [17] C. Gyurik and V. Dunjko, "On establishing learning separations between classical and quantum machine learning with classical data," Aug. 2023, doi: [10.48550/arXiv.2208.06339](https://doi.org/10.48550/arXiv.2208.06339).
- [18] T. Haug, K. Bharti, and M. Kim, "Capacity and quantum geometry of parametrized quantum circuits," *PRX Quantum*, vol. 2, Oct. 2021, Art. no. 040309, doi: [10.1103/PRXQuantum.2.040309](https://doi.org/10.1103/PRXQuantum.2.040309).
- [19] A. M. Childs, J.-P. Liu, and A. Ostrander, "High-precision quantum algorithms for partial differential equations," *Quantum*, vol. 5, Nov. 2021, Art. no. 574, doi: [10.22331/q-2021-11-10-574](https://doi.org/10.22331/q-2021-11-10-574).
- [20] A. J. Pool, A. D. Somoza, M. Lubasch, and B. Horstmann, "Solving partial differential equations using a quantum computer," in *Proc. IEEE Int. Conf. Quantum Comput. Eng.*, 2022, pp. 864–866, doi: [10.1109/QCE53715.2022.00146](https://doi.org/10.1109/QCE53715.2022.00146).
- [21] N. Gourianov et al., "A quantum-inspired approach to exploit turbulence structures," *Nature Comput. Sci.*, vol. 2, no. 1, pp. 30–37, 2022, doi: [10.1038/s43588-021-00181-1](https://doi.org/10.1038/s43588-021-00181-1).
- [22] D-Wave Systems, "D-wave quantum computing," 2023, Accessed: Nov., 2023, doi: [10.1038/s43588-021-00181-1](https://doi.org/10.1038/s43588-021-00181-1).
- [23] C. Gross and I. Bloch, "Quantum simulations with ultracold atoms in optical lattices," *Science*, vol. 357, no. 6355, pp. 995–1001, 2017, doi: [10.1126/science.aal3837](https://doi.org/10.1126/science.aal3837).
- [24] L. Funcke, T. Hartung, K. Jansen, and S. Kühn, "Review on quantum computing for lattice field theory," in *Proc. 39th Int. Symp. Lattice Field Theory*, 2023, Art. no. 228. [Online]. Available: <https://pos.sissa.it/430/228/pdf>
- [25] IBM, "IBM quantum computing," 2023. [Online]. Available: <https://www.ibm.com/quantum>
- [26] A. Acín et al., "The quantum technologies roadmap: A European community view," *New J. Phys.*, vol. 20, no. 8, Aug. 2018, Art. no. 080201, doi: [10.1088/1367-2630/aad1ea](https://doi.org/10.1088/1367-2630/aad1ea).
- [27] S. Aaronson, "How much structure is needed for huge quantum speedups?," 2022, doi: [10.48550/arXiv.2209.06930](https://doi.org/10.48550/arXiv.2209.06930).
- [28] M. Dalmonte, B. Vermersch, and P. Zoller, "Quantum simulation and spectroscopy of entanglement Hamiltonians," *Nature Phys.*, vol. 14, no. 8, pp. 827–831, Aug. 2018, doi: [10.1038/s41567-018-0151-7](https://doi.org/10.1038/s41567-018-0151-7).
- [29] S. Lu, M. C. Bañuls, and J. I. Cirac, "Algorithms for quantum simulation at finite energies," *PRX Quantum*, vol. 2, May 2021, Art. no. 020321, doi: [10.1103/PRXQuantum.2.020321](https://doi.org/10.1103/PRXQuantum.2.020321).
- [30] S. Otgonbaatar et al., "Quantum computing for climate change detection, climate modeling, and climate digital twin," Tech. Rep., Nov. 2023. [Online]. Available: <https://elib.dlr.de/198760/>
- [31] R. Babbush, J. R. McClean, M. Newman, C. Gidney, S. Boixo, and H. Neven, "Focus beyond quadratic speedups for error-corrected quantum advantage," *PRX Quantum*, vol. 2, no. 1, Mar. 2021, Art. no. 010103, doi: [10.1103/prxquantum.2.010103](https://doi.org/10.1103/prxquantum.2.010103).
- [32] M. A. Nielsen, I. Chuang, and L. K. Grover, "Quantum computation and quantum information," *Amer. J. Phys.*, vol. 70, no. 5, pp. 558–559, 2002, doi: [10.1119/1.1463744](https://doi.org/10.1119/1.1463744).
- [33] S. Otgonbaatar and M. Datcu, "Assembly of a coresets of Earth observation images on a small quantum computer," *Electronics*, vol. 10, no. 20, 2021, Art. no. 2482. [Online]. Available: <https://www.mdpi.com/2079-9292/10/20/2482>
- [34] P. Helber, B. Bischke, A. Dengel, and D. Borth, "EuroSAT: A novel dataset and deep learning benchmark for land use and land cover classification," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 7, pp. 2217–2226, Jul. 2019, doi: [10.1109/JSTARS.2019.2918242](https://doi.org/10.1109/JSTARS.2019.2918242).
- [35] R. Acharya et al., "Suppressing quantum errors by scaling a surface code logical qubit," *Nature*, vol. 614, no. 7949, pp. 676–681, Feb. 2023, doi: [10.1038/s41586-022-05434-1](https://doi.org/10.1038/s41586-022-05434-1).
- [36] S. Otgonbaatar and M. Datcu, "Classification of remote sensing images with parameterized quantum gates," *IEEE Geosci. Remote Sens. Lett.*, vol. 19, 2022, Art. no. 8020105, doi: [10.1109/LGRS.2021.3108014](https://doi.org/10.1109/LGRS.2021.3108014).
- [37] P. Gawron and S. Lewiński, "Multi-spectral image classification with quantum neural network," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 3513–3516, doi: [10.1109/IGARSS39084.2020.9323065](https://doi.org/10.1109/IGARSS39084.2020.9323065).
- [38] Y. Yang and S. Newsam, "Bag-of-visual-words and spatial extensions for land-use classification," in *Proc. 18th SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, 2010, pp. 270–279, doi: [10.1145/1869790.1869829](https://doi.org/10.1145/1869790.1869829).
- [39] S. Otgonbaatar, G. Schwarz, M. Datcu, and D. Kranzlmüller, "Quantum transfer learning for real-world, small, and high-dimensional remotely sensed datasets," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 16, pp. 9223–9230, 2023, doi: [10.1109/JSTARS.2023.3316306](https://doi.org/10.1109/JSTARS.2023.3316306).
- [40] S. Otgonbaatar and M. Datcu, "Natural embedding of the stokes parameters of polarimetric synthetic aperture radar images in a gate-based quantum computer," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2022, Art. no. 4704008, doi: [10.1109/TGRS.2021.3110056](https://doi.org/10.1109/TGRS.2021.3110056).
- [41] H.-Y. Huang et al., "Power of data in quantum machine learning," *Nature Commun.*, vol. 12, no. 1, May 2021, Art. no. 2631, doi: [10.1038/s41467-021-22539-9](https://doi.org/10.1038/s41467-021-22539-9).
- [42] M. K. Gupta, M. Beseda, and P. Gawron, "How quantum computing-friendly multispectral data can be?," in *Proc. Int. Geosci. Remote Sens. Symp.*, 2022, pp. 4153–4156, doi: [10.1109/IGARSS46834.2022.9883676](https://doi.org/10.1109/IGARSS46834.2022.9883676).
- [43] E. Boyda, S. Basu, S. Ganguly, A. Michaelis, S. Mukhopadhyay, and R. R. Nemani, "Deploying a quantum annealing processor to detect tree cover in aerial imagery of California," *PLoS One*, vol. 12, no. 2, pp. 1–22, 2017, doi: [10.1371/journal.pone.0172505](https://doi.org/10.1371/journal.pone.0172505).
- [44] S. Otgonbaatar and M. Datcu, "Quantum annealing approach: Feature extraction and segmentation of synthetic aperture radar image," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 3692–3695, doi: [10.1109/IGARSS39084.2020.9323504](https://doi.org/10.1109/IGARSS39084.2020.9323504).
- [45] D. Willsch, M. Willsch, H. De Raedt, and K. Michielsen, "Support vector machines on the D-wave quantum annealer," *Comput. Phys. Commun.*, vol. 248, 2020, Art. no. 107006, doi: [10.1016/j.cpc.2019.107006](https://doi.org/10.1016/j.cpc.2019.107006).

- [46] G. Cavallaro, M. Riedel, M. Richerzhagen, J. A. Benediktsson, and A. Plaza, "On understanding Big Data impacts in remotely sensed image classification using support vector machine methods," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 8, no. 10, pp. 4634–4646, Oct. 2015, doi: [10.1109/JSTARS.2015.2458855](https://doi.org/10.1109/JSTARS.2015.2458855).
- [47] S. Otgonbaatar, M. Datcu, and B. Demir, "Coreset of hyperspectral images on a small quantum computer," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2022, pp. 4923–4926, doi: [10.1109/IGARSS46834.2022.9884273](https://doi.org/10.1109/IGARSS46834.2022.9884273).
- [48] S. Otgonbaatar and M. Datcu, "A quantum annealer for subset feature selection and the classification of hyperspectral images," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 14, pp. 7057–7065, 2021, doi: [10.1109/JSTARS.2021.3095377](https://doi.org/10.1109/JSTARS.2021.3095377).
- [49] J. Chen, E. Stoudenmire, and S. R. White, "Quantum fourier transform has small entanglement," *PRX Quantum*, vol. 4, Oct. 2023, Art. no. 040318, doi: [10.1103/PRXQuantum.4.040318](https://doi.org/10.1103/PRXQuantum.4.040318).
- [50] E. M. Stoudenmire and D. J. Schwab, "Supervised learning with quantum-inspired tensor networks," 2016, Accessed: Nov., 2023, doi: [10.48550/arXiv.1605.05775](https://doi.org/10.48550/arXiv.1605.05775).
- [51] Z.-F. Gao et al., "Compressing deep neural networks by matrix product operators," *Phys. Rev. Res.*, vol. 2, Jun. 2020, Art. no. 023300, doi: [10.1103/PhysRevResearch.2.023300](https://doi.org/10.1103/PhysRevResearch.2.023300).
- [52] F. Verstraete, T. Nishino, U. Schollwöck, M. C. Bañuls, G. K. Chan, and M. E. Stoudenmire, "Density matrix renormalization group, 30 years on," *Nature Rev. Phys.*, vol. 5, pp. 273–276, Apr. 2023, doi: [10.1038/s42254-023-00572-5](https://doi.org/10.1038/s42254-023-00572-5).
- [53] H. Huang, X.-Y. Liu, W. Tong, T. Zhang, A. Walid, and X. Wang, "High performance hierarchical tucker tensor learning using GPU tensor cores," *IEEE Trans. Comput.*, vol. 72, no. 2, pp. 452–465, Feb. 2023, doi: [10.1109/TC.2022.3172895](https://doi.org/10.1109/TC.2022.3172895).
- [54] S. Otgonbaatar and D. Kranzlmüller, "Quantum-inspired tensor network for Earth science," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2023, pp. 788–791, doi: [10.1109/IGARSS52108.2023.10282577](https://doi.org/10.1109/IGARSS52108.2023.10282577).
- [55] F. Fan, Y. Shi, T. Guggemos, and X. X. Zhu, "Hybrid quantum-classical convolutional neural network model for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Sep. 18, 2023, doi: [10.1109/TNNLS.2023.3312170](https://doi.org/10.1109/TNNLS.2023.3312170).
- [56] M. C. Caro et al., "Generalization in quantum machine learning from few training data," *Nature Commun.*, vol. 13, no. 1, Aug. 2022, Art. no. 4919, doi: [10.1038/s41467-022-32550-3](https://doi.org/10.1038/s41467-022-32550-3).
- [57] J. Preskill, "Quantum computing in the NISQ era and beyond," *Quantum*, vol. 2, 2018, Art. no. 79, doi: [10.22331/q-2018-08-06-79](https://doi.org/10.22331/q-2018-08-06-79).
- [58] S. Bravyi and D. Gosset, "Improved classical simulation of quantum circuits dominated by clifford gates," *Phys. Rev. Lett.*, vol. 116, Jun. 2016, Art. no. 250501, doi: [10.1103/PhysRevLett.116.250501](https://doi.org/10.1103/PhysRevLett.116.250501).
- [59] S. Aaronson and D. Gottesman, "Improved simulation of stabilizer circuits," *PhysRevA*, vol. 70, no. 5, Nov. 2004, doi: [10.1103/PhysRevA.70.052328](https://doi.org/10.1103/PhysRevA.70.052328).
- [60] J. Tindall, M. Fishman, M. Stoudenmire, and D. Sels, "Efficient tensor network simulation of IBM's kicked Ising experiment," 2023, doi: [10.48550/arXiv.2306.14887](https://doi.org/10.48550/arXiv.2306.14887).
- [61] D. Litinski, "A game of surface codes: Large-scale quantum computing with lattice surgery," *Quantum*, vol. 3, Mar. 2019, Art. no. 128, doi: [10.22331/q-2019-03-05-128](https://doi.org/10.22331/q-2019-03-05-128).
- [62] M. Hinsche et al., "One T gate makes distribution learning hard," *Phys. Rev. Lett.*, vol. 130, Jun. 2023, Art. no. 240602, doi: [10.1103/PhysRevLett.130.240602](https://doi.org/10.1103/PhysRevLett.130.240602).
- [63] A. Mari, T. R. Bromley, J. Izaac, M. Schuld, and N. Killoran, "Transfer learning in hybrid classical-quantum neural networks," *Quantum*, vol. 4, Oct. 2020, Art. no. 340, doi: [10.22331/q-2020-10-09-340](https://doi.org/10.22331/q-2020-10-09-340).
- [64] M. E. Beverland et al., "Assessing requirements to scale to practical quantum advantage," 2022, doi: [10.48550/arXiv.2211.07629](https://doi.org/10.48550/arXiv.2211.07629).
- [65] G. Cheng, J. Han, and X. Lu, "Remote sensing image scene classification: Benchmark and state of the art," *Proc. IEEE*, vol. 105, no. 10, pp. 1865–1883, Oct. 2017, doi: [10.1109/JPROC.2017.2675998](https://doi.org/10.1109/JPROC.2017.2675998).
- [66] M. Paoletti, J. Haut, J. Plaza, and A. Plaza, "Deep learning classifiers for hyperspectral imaging: A review," *ISPRS J. Photogrammetry Remote Sens.*, vol. 158, pp. 279–317, 2019, doi: [10.1016/j.isprsjprs.2019.09.006](https://doi.org/10.1016/j.isprsjprs.2019.09.006).
- [67] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, doi: [10.48550/arXiv.1512.03385](https://doi.org/10.48550/arXiv.1512.03385).
- [68] E. Farhi and H. Neven, "Classification with quantum neural networks on near term processors," 2018, doi: [10.48550/arXiv.1802.06002](https://doi.org/10.48550/arXiv.1802.06002).
- [69] R. Fok, A. An, and X. Wang, "Spontaneous symmetry breaking in neural networks," 2017, doi: [10.48550/arXiv.1710.06096](https://doi.org/10.48550/arXiv.1710.06096).
- [70] A. G. Fowler and C. Gidney, "Low overhead quantum computation using lattice surgery," 2019, doi: [10.48550/arXiv.1808.06709](https://doi.org/10.48550/arXiv.1808.06709).
- [71] C. Gidney, "Stim: A fast stabilizer circuit simulator," *Quantum*, vol. 5, Jul. 2021, Art. no. 497, doi: [10.22331/Fq-2021-07-06-497](https://doi.org/10.22331/Fq-2021-07-06-497).
- [72] M. Reiher, N. Wiebe, K. M. Svore, D. Wecker, and M. Troyer, "Elucidating reaction mechanisms on quantum computers," *Proc. Nat. Acad. Sci.*, vol. 114, no. 29, pp. 7555–7560, 2017, doi: [10.1073/pnas.1619152114](https://doi.org/10.1073/pnas.1619152114).
- [73] M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, "Circuit-centric quantum classifiers," *Phys. Rev. A*, vol. 101, no. 3, Mar. 2020, doi: [10.1103/PhysRevA.101.032308](https://doi.org/10.1103/PhysRevA.101.032308).
- [74] M. Benedetti, B. Coyle, M. Fiorentini, M. Lubasch, and M. Rosenkranz, "Variational inference with a quantum computer," *Phys. Rev. Appl.*, vol. 16, Oct. 2021, Art. no. 044057, doi: [10.1103/PhysRevApplied.16.044057](https://doi.org/10.1103/PhysRevApplied.16.044057).

5.1.3 Publication I: Quantum Computing for Climate Change Detection, Climate Modeling, and Climate Digital Twins

S. Otgonbaatar, O. Nurmi, M. Johansson, J. Mäkelä, T. Kocman, P. Gawron, Z. Puchala, J. Mielczarek, A. Miroszewski, and C.O. Dumitru

Abstract. This work studies the potential of [QC](#) and [QML](#) models for detecting climate change, climate modeling, and climate digital twins. We also compare the time and energy consumption of some quantum computers and a classical computer. Additionally, we identify a few use cases which are difficult for a conventional computer but can be tackled with quantum computers or by integrating them in [HPC](#) systems. We evaluate the effectiveness of quantum annealers, quantum simulators, and universal quantum computers, each designed to solve specific types of computational problems that are typically challenging.

Quantum Computing for Climate Change Detection, Climate Modeling, and Climate Digital Twin

Soronzonbold Otgonbaatar^{1,2}, Olli Nurmi³, Mikael Johansson⁴,
Jarmo Mäkelä⁴, Tadeusz Kocman⁵, Piotr Gawron⁶,
Zbigniew Puchala⁶, Jakub Mielczarek⁸, Artur Miroszewski⁸,
Corneliu Octavian Dumitru¹

^{1*}Remote Sensing Technology Institute, German Aerospace Center
DLR, Weßling, 82234, Germany.

²Institut für Informatik, Ludwig-Maximilians-Universität München,
Oettingenstr. 67, München, 80538, Germany.

³Microelectronics and quantum technology, VTT-Technical Research
Centre of Finland Ltd, Tietotie 3, Espoo, 02150, Finland.

⁴Advanced Computing Facility, CSC-IT Center for Science Ltd,
Keilaranta 14, Espoo, 02150, Finland.

⁵Business Development Department, AROBS Polska sp. z o.o., Trzy
Lipy 3B/3.11.5, Gdansk, 80-172, Poland.

⁶Scientific Computing and Information Technology Group, AstroCeNT,
Nicolaus Copernicus Astronomical Center of the Polish Academy of
Sciences, ul. Rektorska 4, Warsaw, 00-614, Poland.

⁸Institute of Theoretical Physics, Jagiellonian University, Łojasiewicza
11, Cracow, 30-348, Poland.

Contributing authors: soronzonbold.otgonbaatar@dlr.de;
olli.nurmi@vtt.fi; mikael.johansson@csc.fi; jarmo.makela@csc.fi;
tadeusz.kocman@arobs.pl; gawron@camk.edu.pl;
zbigniew.puchala@centrumetos.pl; jakub.mielczarek@uj.edu.pl;
artur.miroszewski@uj.edu.pl; corneliu.dumitru@dlr.de;

Abstract

This study assesses and examines both quantum machine learning and quantum approaches for climate change detection, climate modeling, and climate digital twin, whereas we consider the time and energy consumption of quantum machines and a classical computer. We identified several use-case instances for climate change detection, climate modeling, and climate digital twin that are notoriously difficult for a conventional computer but can be tackled efficiently using quantum machines or quantum machines integrated with a conventional computer. We also assessed quantum machines divided into a quantum annealer, a quantum simulator, and a universal quantum computer, each of which proposes to solve specific classes and forms of intractable computational problems.

Keywords: quantum machine learning, quantum computer, high-performance computing, quantum resource estimation, climate change detection, climate modeling, climate digital twin, Earth observation, remote sensing, hyperspectral images, image analysis.

1 Introduction

Quantum computing is a novel computing paradigm that processes digital information based on quantum mechanical principles in contrast to conventional classical computing. Quantum machines using primitives of quantum computing, in principle, promise to generate better and faster solutions to some inherently hard computational problems [1]; the hardness of computational problems refers to time and memory-space measures in computational complexity theories/conjectures required for finding their solutions. Some quantum machines are even known to utilize less electrical power compared to conventional supercomputers. For example, a D-Wave quantum annealer consumes 25 KW power, whereas the Summit supercomputer consumes 13 MW power [2]. Based on the time and memory-space measures, computational problems are classified according to their hardness (see Fig. 1). Intractable computational problems are ubiquitous in space and the aerospace industry. Examples of hard problems include resource allocation, planning, object scheduling, and Artificial Intelligence (AI) model training while considering time, memory space, and electrical consumption. Hence, there already exist some quantum approaches for real-world intractable computational problems in the aerospace industry, e.g., a flight-gate assignment [3], satellite mission planning for Earth observation [4], numerical weather modeling and climate simulation involving Partial Differential Equations (PDEs) [5], energy optimization and a renewable energy sector [6, 7], and quantum AI for climate change detection [8, 9]. However, there is still no demonstrated quantum advantage for tackling practical problems over conventional classical techniques. In particular, quantum machines are in their fancy, and it is not well known which practical problem will inherently profit from quantum machines or which quantum machine will meet dead-end. There is an ongoing effort to identify hard computational problems in the space and aerospace industry that can be tackled more efficiently using quantum machines than the supercomputer or how to profit from both quantum machines and supercomputer [8]. Therefore, we

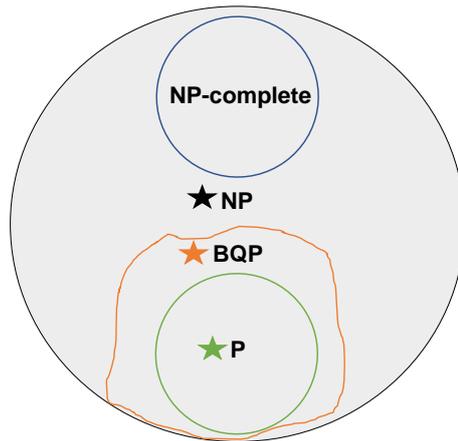


Fig. 1 Computational complexity for computational problems. Here, the orange star denotes a class of computational problems which is hard for a classical computer but easy for a quantum computer, or a polynomial-depth quantum algorithm exists. Here, **NP** stands for non-deterministic polynomial time problems, **P** stands for polynomial time problems, and **BQP** stands for bounded-error quantum polynomial problems. Taken from Fig. 1 of the article [8].

assess different quantum machines and provide their performance-related parameters while considering their time and electric power consumption. We also identify some climate-related use-cases that are inherently hard for supercomputing systems but could be tackled using quantum machines or supercomputers integrated with quantum machines.

2 The assessment of quantum technology

The development of quantum computing encompasses a wide range of technologies, from hardware systems to software tools depicted in Fig. 2. The quantum computing industry is still in its infancy and, like the early days of classical computing, without well-defined interfaces between the various parts of the quantum computer. The quality of a quantum algorithm is affected not only by the quality of the individual constituent components (qubits, gates, measurements) but also by the interplay of global device and algorithmic properties such as device topology, multi-qubit noise correlations, and circuit structures. Also, the quantum compilers and middleware affect the algorithm performance to be run on certain hardware. Typically, the machine instructions are optimized for execution on all hardware platforms. After the execution, additional postprocessing may also be employed to improve readout efficiency. These optimizations typically include:

1. depth reduction and logical transpilation: A sequence of compiler passes is used to mathematically reduce the gate depth (e.g., T-gate count) of the quantum circuit and the logical operations in the circuit are mapped to the native gates available on the hardware.
2. error-aware hardware mapping: Error-aware compilation is used to best select the appropriate subset and logical assignment of qubits on a device.

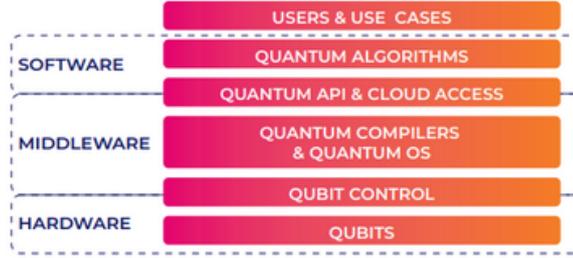


Fig. 2 Quantum stack by European Quantum Industry Consortium (QuIC) showing the software, middleware, and hardware layers that have direct impact on the use cases and their prospects [11].

3. elimination of circuit crosstalk: Dynamical decoupling sequences are incorporated to mitigate various idling errors, including dephasing and ZZ crosstalk at the algorithmic level.
4. optimized gate replacement: The process involves automated parsing of the device topology to ensure parallel gate optimizations do not share qubits, and relevant single and or multi-qubit gates are optimized.

QC's usefulness heavily depends upon the achievable fidelities and the number of qubits of the Quantum Processing Unit (QPU). Scaling the quality and number of qubits will require advanced 3D architectures and assembly techniques. Some estimates say that achieving practical quantum advantage requires running millions of parallel high-fidelity gates at high speed and reading out millions of qubits in parallel. With current error-correction overheads, practical quantum advantage will be achieved, albeit only for algorithms with small I/O requirements and super quadratic (ideally exponential or quartic) speedups over their classical counterparts [10].

In the current noisy intermediate-scale quantum (NISQ) era, the computation results are mostly limited by errors in single- and two-qubit quantum gates. To succeed roughly half the time in a 100-qubit circuit of depth five, one needs at least 99.9% gate fidelity. In practice, the number of qubits and especially the gate depth required for useful NISQ advantage is likely higher, leading to a fidelity target of 99.99% for all quantum gates, not yet demonstrated. Producing commercially viable QCs requires technologies that facilitate scalable manufacturing, requiring manufacturing process efficiency, reliability, integration, and packaging. Due to manufacturing variability, some qubits may not be functional and available for use; the exact number of qubits yielded will vary with each specific processor manufacturer. The enabling hardware that connects to the QPUs, such as cryogenic coolers, electronic systems, and cabling, must also be matured. The widely accepted approach to remedy the effects of noise and decoherence in quantum computers is using quantum error correction (QEC) [12]. While the hardware requirements to implement fault-tolerant (FT) quantum algorithms have not been met yet, the steady progress in the development of quantum hardware has initiated the introduction of a set of techniques that we refer to broadly as quantum error mitigation. These techniques immediately translate advances in qubit coherence, gate fidelities, readout precision, and speed to measurable advantage in computation. Quantum error mitigation offers the continuous path that will

take us from today’s quantum hardware to tomorrow’s FT quantum computers. They might even be applicable to enable near-term practical quantum advantage without using QEC for certain use cases. A major use case for near- to medium-term quantum computers is accelerating existing HPC workflows. For this, tight integration between HPC and QC, beyond cloud access or the operation as separate computing systems is critical to avoid idle time either due to resource allocation or communications latency. Following three current trends can be identified: (1) stay at ”small” scales (below 100 qubits) and try to solve coherence problems and create useful applications before scaling up; (2) go for large scales (over 1,000 qubits) and try to implement quantum error correction for quantum advantage or superiority while scaling up; (3) scale up and solve large-scale hardware (HW) and software (SW) integration at systems levels.

We mentioned in the abstract that QC hardware could be characterized by the kinds of computation they can run into three categories:

1. **Annealers.** Quantum annealers are a kind of analog quantum simulator relying on the adiabatic theorem and mimicking an Ising Hamiltonian to solve quadratic unconstrained binary optimization (QUBO) problems such as satisfiability and combinatorial search problems. QUBO problems are solved by finding their global minimum over a given set of candidate solutions (candidate states) using quantum fluctuations. In adiabatic computing, noise- and error-tolerance are higher, and it is hard to create entangled states, the main resource for quantum computational advantage over a conventional classical computer.
2. **Analog Quantum Simulators.** Analog quantum simulators are special-purpose devices designed to study quantum systems in a programmable fashion. They exploit superposition and entanglement to provide insight into specific physics problems mimicking the Hamiltonian evolution of the system. Analog quantum simulators are especially suited for simulating quantum physical systems, also, more general optimization is possible. As the quantum interactions between quantum particles are a built-in feature of quantum simulators, near-term quantum advantage is expected for the specific class of problems they can describe.
3. **Digital Universal Quantum Computers, or fault-tolerant universal quantum computers.** The most powerful class of quantum machines that directly exploit superposition, entanglement, and wave-function interference and run quantum algorithms in a step-by-step procedure. In principle, a digital universal quantum computer can solve some computable problems, with the additional advantage of up to exponential speed-up over classical computers. Digital quantum computers operate using quantum gates, logical operations on the basic quantum information primitives. These units are usually two-state quantum bits (qubits), but also continuous-variable (CV) approaches are under development. Qubits can be implemented using several different technologies, e.g., superconducting, trapped ions, neutral atoms, or photonics, which all come with their unique strengths and weaknesses. There are some differences in algorithms between discrete and continuous quantum states, with CV approaches especially suited for, e.g., sampling and regression tasks.

3 The qubit implementation techniques

Plenty of approaches exist to develop scalable qubits with acceptable coherence time and error rate. Some approaches are on a very low TRL level, and estimating their potential is difficult. This chapter describes the six most promising approaches based on published information [13]. The connectivity of a quantum gate processor impacts the depth of actual quantum circuits. During transpilation, an input quantum circuit is compiled to a sequence of native gates or universal gate set such that all operations agree with a specific quantum processor's qubit topology and noise properties. The signal-to-noise ratio impacts on the number of shots required to get a correct answer by recovering the signal. By increasing the gate fidelity a little bit, the number of shots and runtime of a given algorithm may decrease drastically. Even a relatively modest 0.16 percentage point improvement in fidelity could mean it runs in less than half the time. Building large circuits requires long coherent times of the qubit, strong interqubit interaction for fast and high-fidelity two-qubit gates, and small to zero coupling between qubits when no interaction is needed. Transmon qubits allow for various coupling concepts with various pros and cons. Two of the most promising technologies are superconductors and ion traps. At the time of writing, at most 433 and 20 qubits are available for superconducting and ion trap devices, respectively, that is, the IBM Osprey processor, USA and the AQT PINE processor, Austria. And at most 5627 qubits for quantum annealing devices, i.e., D-Wave Advantage. According to the roadmap in 2023, the Advantage 2™ quantum system will incorporate a new qubit design that enables 20-way connectivity in a new topology containing 7000+ qubits and make use of the latest improvements in quantum coherence in a multi-layer fabrication stack (see Fig. 3).

1. **Superconducting circuits.** Physical implementations of superconducting qubits reside on the chip at fixed locations and are connected via a well-defined pattern, the so-called connectivity structure. Structures are designed to minimize the possibility of frequency collisions and optimize the hardware performances. The larger the number of neighbors of a qubit, the more frequencies are required to realize two-qubit gates using cross-resonance interaction. Current technology can turn off the coupling of transmon qubits with close frequencies, but this is prone to crosstalk errors. A more efficient pulse shape could be optimized with tunable couplers to achieve a CZ gate with a higher fidelity and lower unwanted leakage. Until recently, the mainstay devices have been fixed couplers with a constant coupling strength. However, attention is now turning to tunable couplers, which are seen as offering the adjustable coupling strength necessary to improve performance. Roadmaps aim for increased coherence, yield, and reproducibility, enabling higher gate fidelity and, consequently, larger circuit depth on an equal footing with increased qubit number. Three-dimensional multi-chips allow massive scaling of QPUs. It is also necessary to reduce variation of all critical parameters and tolerances for all steps of chip fabrication and 3D integration. Chip engineering needs to consider signal routing, the electromagnetic environment, quantum coherence, and robustness against variations in device parameters. The advanced state of the art in quantum-processor performance requires the development of novel components for fast and

highly selective multiplexed readout, elements for mid-circuit leakage detection, coupling schemes to accelerate parity measurements, conditional and unconditional reset capabilities, and highly parallelizable two-qubit gates. Ramp-up and operating large-scale QPU also requires advancing the room-temperature electronic (RTE) systems with sufficient control and readout channels and capability for real-time quantum error correction.

2. **Trapped ions.** Ion traps use ions, single-charged atoms, as qubits. Information is encoded in the electronic state of ions that are confined using electric fields. Operations are performed with tailored laser pulses that modify the state of the ions. Ion-trap quantum computers provide optical interfaces and high-fidelity local operations. Multiple ion-trapping potentials can be connected deterministically by physically transporting ions across micro-scale segmented ion traps, which forms an architecture for a scalable quantum information processor. Realizing trapped-ion qubits requires the orchestration of several devices, including the ion source, dedicated lasers, several optical components and sensors, a vacuum, cooling mechanisms, and control and measurement electronics. The respective systems routinely operate with about 20-30 qubits but can be pushed (at reduced levels of control) up to 50 qubits. The devices hold fully connected quantum registers, which facilitate the implementation of quantum algorithms. For trapped ion qubits, the main noise is not relaxation with time T_1 but instead dephasing with time T_2 induced by fluctuation of magnetic fields. Also, the state-detection efficiency decreases with the motional heating of the ion without laser cooling.
3. **Photonic.** Qubits are realized by processing states of different modes of light through both linear and nonlinear elements. The fundamental building blocks include deterministic single-photon sources, integrated photonic circuits, and efficient single-photon detectors. Photonic systems have the unique property that they can operate at room temperature and allow for easy transfer of quantum information. The main disadvantage of photonic systems is that performing a precise interaction between photons is a difficult task to achieve. In recent years, a couple of programmable and scalable architectures for photonic quantum computing were introduced, and specific quantum algorithms such as Gaussian boson sampling, molecular vibronic spectra, and graph similarity were executed in laboratories. Due to photons' properties, photonic circuits have different features from qubit-based systems from the point of view of computing and operations.
4. **Neutral atoms.** Qubits are realized by internal states of neutral atoms trapped in an optical lattice. Like ion-trap systems, qubits can be programmed using the energy levels of the atoms. Light, or electromagnetic radiation, can be used to trap and manipulate the quantum states of uncharged (neutral) atoms. Multiple qubits that are nearby in space can be programmed to interact with one another via two-qubit gates. This opens new possibilities for exotic quantum-computing circuit topologies. Neutral atom platforms for quantum processing have a unique potential for scalability: the size of the quantum register is only limited by the amount of trapping laser power and by the performance of the optical system generating the optical tweezers.

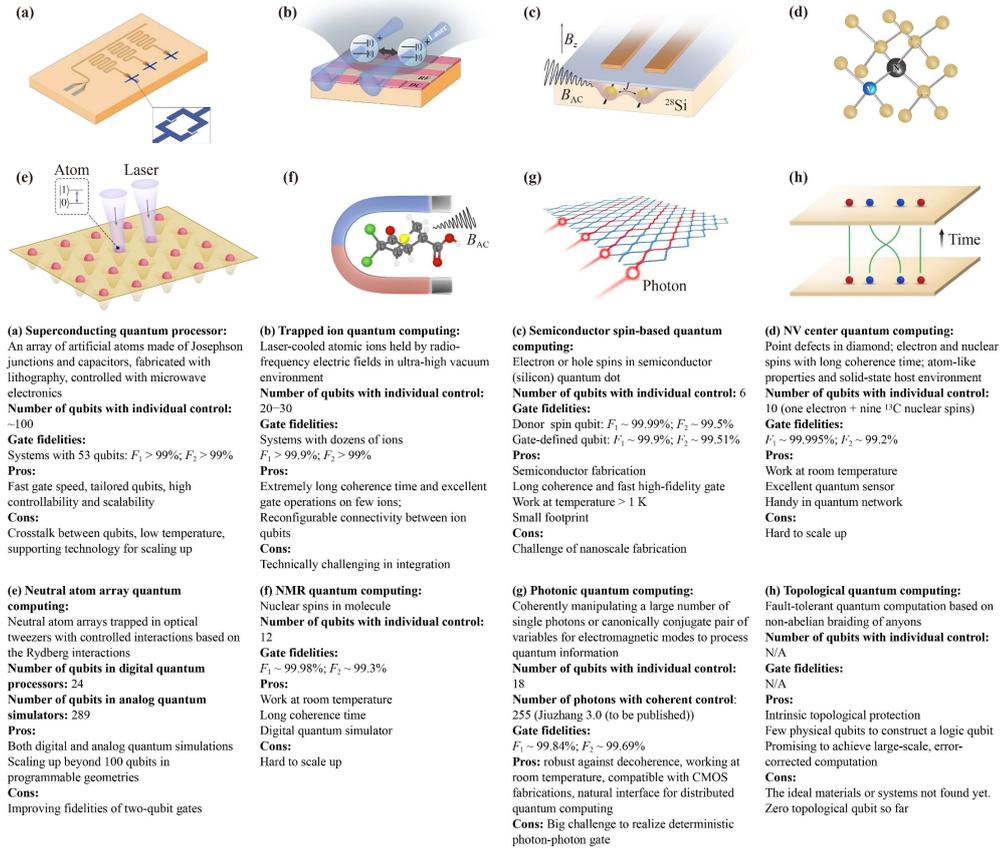


Fig. 3 Reproduction of Fig. 2 from [14] presenting a selection of quantum computing hardware. (CC-BY 4.0)

- Silicon spin.** QPU integrates both qubits and control electronics and operates at a liquid helium temperature (4K), which is higher than the usual millikelvin temperatures of superconducting qubit systems. The higher operating temperatures result in lower quality qubits but extensive and efficient control electronics.
- NV diamond.** Qubits are realized by the electronic or nuclear spin of nitrogen-vacancy centers in diamond. In these artificial diamond structures, a carbon atom has been replaced by a nitrogen atom near a carbon atom gap. Qubit gates are implemented with microwaves, a magnetic field, and an electric field. Qubit readout uses a laser and fluorescence detection.

3.1 QPU performance consideration

To implement a functional quantum computer requires an integrated system consisting of a quantum processor, its fabrication, packaging and wiring, room temperature

electronics, enabling software, system integration, application development, and testing system. Increasing QPU performance means improving all the subsystems and subcomponents of the machine individually and simultaneously while ensuring all the systems continue to work well together. Here, we focus on the Quantum Processing Unit, QPU. There is not yet a standard to assess the performance levels of the processor. Some approaches include benchmarking metrics such as Quantum volume, Algorithmic volume, and Randomised benchmarking. To keep the qubit error rates below a certain threshold for fault-tolerant computation, extending the coherence time of qubits is crucial. Here we list some current critical areas in Qubit implementation, Qubit control, Qubit calibration, and Code running.

Currently, only trapped ions and superconducting qubits satisfy the five required criteria for quantum computing defined by DiVincenzo [15]:

1. A scalable physical system with well-characterized qubits;
2. The ability to initialize the state of the qubits to a simple fiducial state;
3. Long relevant decoherence times;
4. A “universal” set of quantum gates;
5. A qubit-specific measurement capability.

Typical physical indicators of quantum computers include T_1 , T_2 , single-qubit gate fidelity, two-qubit gate fidelity, and readout fidelity. The aggregated benchmarks can help the user to determine the performance of a quantum processor with only one or several parameters. The aggregated metrics can be calculated with randomly generated quantum circuits or estimated based on the basic physical properties of a quantum processor. Typical aggregated benchmarks include quantum volume (QV) and algorithmic qubits (AQ). Specific attributes (e.g., faster gate speeds, higher fidelities, denser connectivity) can make certain machines better at particular tasks than others but not superior in absolute terms for the time being. Also, certain QPU may then fit better on certain QC4EO use cases based on the problem they need to solve instead of an arbitrary rating. For example, ion trap devices are able to make up for slower operation speeds with better connectivity. In contrast, superconducting systems with sparser connectivities are still competitive due to their much faster operation times. Another point is that trapped ion qubits have very long coherence times, making them more robust to mid-circuit measurement — a key requirement for error correction. However, the 1000x faster gate speeds of superconducting are preferable for variational benchmarks like QAOA, which require millions of sequential iterations [13]. We presented example quantum machines in the Table 1.

3.2 Sizing QPUs

Modern classical Central Processing Units (CPUs) operate at around 3GHz clock cycle speed, or around 0.30ns clock cycle time. Nowadays, hard computational problems are even tackled on several hundreds of parallel CPUs as well as General Processing Units (GPUs). The fastest QPU is currently a superconducting-based QPU (see below tables) in terms of the qubit and quantum gate operation time, that is, clock cycle time. However, I/O speed is 10,000 slower in QPU compared to CPU. Logical qubit/magic state distillation (creating more accurate quantum states from multiple noisy ones)

Table 1 Some dominating quantum machines in the global market are offered by large organizations. See Fig. 4 for the projection of the roadmap of some quantum machines and Table 2 for parameters of quantum machines [16].

Organizations	Locations	Technology	Current qubits	Projected qubits (3-5 years)
IBM	USA	superconducting	433	4, 158
Google	USA	superconducting	73	100
IQM	FI	superconducting	20	54
USTC	CN	superconducting	66	100
AQT	AT	trapped ions	20	200
IONQ	USA	trapped ions	29	256
Xanadu	CA	photonic	216	216
USTC	CN	photonic	113	300
D-Wave	CA	superconducting-annealing	5, 000	10, 000
QuEra	USA	neutral atoms	256	1, 000

Table 2 Sizing quantum machines: SC—superconducting QCs [17], T.ions—trapped ions QCs [18], N.atoms—neutral atoms QCs [19], Photonic—photonic QCs [20], S.spin—silicon spin QCs [21], NV—nitrogen vacancy in diamond QCs [13], CPUs—conventional central processing units. See also the Table 1.

Parameters	SC	T.ions	Photonic	N.atoms	S.spin	NV	CPUs
Clock cycle	1MHz	1KHz	10Hz	1MHz	0.76MHz	1MHz	3GHz
Measurement	660ns	300μs	x	200ms	1.3μs	x	x
2-qubit gate	34ns	200μs	x	< 100μs	x	700ns	x
1-qubit gate	25ns	15μs	x	x	x	9ns	x
Readout fidelity	99.4%	97.3%	50.0%	99.1%	99%	98%	x
1Q fidelity	99.99%	99.99%	99.84%	99.83%	99.99%	99.99%	x
2Q fidelity	99.97%	99.9%	99.69%	99.4%	99.5%	99.2%	x

is another restriction, and another restriction is high-bandwidth, low-noise classical electronics. Hence, to beat CPUs, there is a need to improve the speed of the whole I/O system in QPUs from register preparation to read-out. More than exponential speedup is also required in the quantum algorithm [10], and only some of the problems are meaningful to compare depending on their parallelizability on CPUs and GPUs (see Table 2 and Fig. 4). Regardless of the qubit technology, there is the persisting challenge to scale logical error-free qubits due to the quantum state generation having a high fidelity and classical electronics controls, to name a few [13].

3.3 Error mitigation and correction

Errors are generated by various interactions, electromagnetic or mechanical, between qubits and their immediate environment and are associated with the phenomenon of

quantum decoherence. Error removal is progressing steadily but barely managing to gain one or two orders of magnitude in error, whereas in an ideal world, we would need ten orders of magnitude improvements. It is possible to correct errors, even by using noisy gates, provided that the noise level remains below a certain threshold. The drawback is that it requires a huge overhead of physical qubits and classical information processing (see Fig. 5) [12]. There is an optimal “code size”, i.e., a number of physical qubits per logical qubit, that maximizes the metric of performance— and beyond which more error correction degrades the computation accuracy rates. Also, less noise mandates a bigger code and more physical qubits, but more physical qubits give rise to more heat generation, hence more noise. To execute a quantum application successfully, QEC must be used to build logical qubits that can be used to store and manipulate quantum information better than raw physical qubits. This QEC capability is central to scalable quantum computers, but the costs are formidable, often multiplying the number of qubits needed by a factor of thousands and runtimes by a factor of hundreds. One of the trends for improving the error correction rate characteristics is employing AI models for this process [22]. This would in turn allow us to reduce the number of quantum computation instances needed before obtaining a reliable result or decrease the number of physical qubits in QC systems. In Europe, there exists a start-up that develops a toolkit for providing this form of QC improvement [23]. An important metric for a QEC approach is its threshold, which specifies the maximum error rate that it can tolerate. Physical error rates on Clifford operations below 0.1% (including qubit preparations, measurements, and gates) are typically required to avoid prohibitive QEC overheads. These values are possible to obtain only in the setting where operations can be applied in parallel, which may pose a significant hardware challenge for some platforms, such as trapped ions. In many QEC schemes, the non-Clifford gates (typically T gate) are quite costly when requiring fault tolerance [24]. The required low error rate T states are produced using a T state distillation factory involving a sequence of rounds of distillation, where each round takes in many noisy T states encoded in a smaller distance code, processes them using a distillation unit, and outputs fewer less noisy T states encoded in a larger distance code, with the number of rounds, distillation units, and distances all being parameters which can be varied. This procedure is iterated, where the output T states of one round are fed into the next round as inputs. T factories incur significant physical overheads, requiring several thousand physical qubits and only producing new T states once every 10 to 15 logical time steps [25].

Microsoft (MS) has evaluated three use cases concluding that to achieve practical quantum advantage QC’s need to be able to control millions of parallel operations with low error rates, and to read out those millions of qubits in parallel to enable decoding of the errors at speed, all while ensuring the overarching logical clock time is fast enough to complete the computation within a month runtime or less [26]. MS concluded that logical gate times under $10\mu s$, in turn requiring physical gate times around $100ns$, would be needed to complete the quantum chemistry algorithm within a month, using a few million physical qubits. To execute syndrome measurements on these qubits and communicating the quantum measurements to the decoder requires a large quantum-classical bandwidth and processing power for decoding. The exact

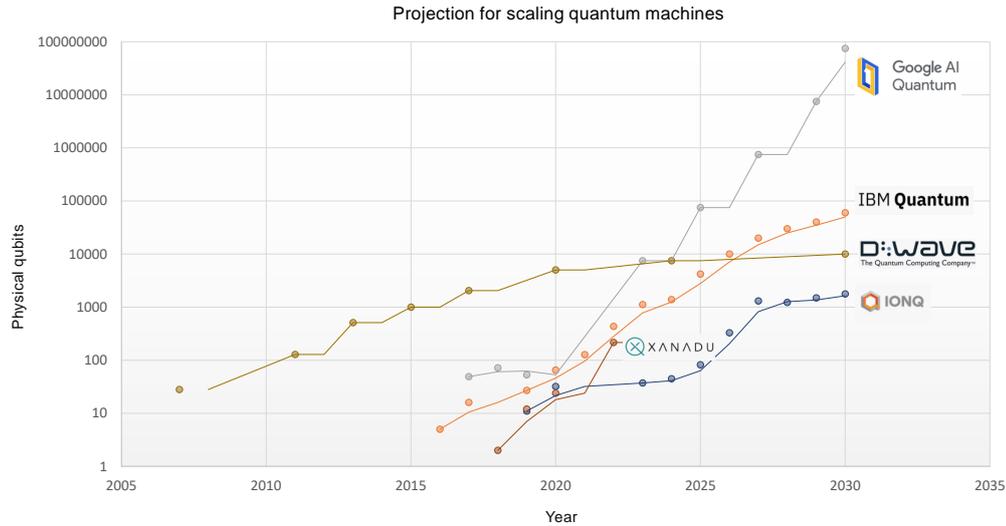


Fig. 4 Quantum machines roadmap of some organizations which provide the open-data for their quantum development projection [speculation].

estimates of bandwidth requirements depend on the choice of QEC code, system size and physical operation times, but roughly, with a few million qubits, the estimation is that several terabytes per second of bandwidth will be required between the quantum and classical planes. Furthermore, processing these measurements at a rate that is sufficient to effectively correct errors, demand petascale classical computing resources that are tightly integrated with the quantum machine.

4 Investment in quantum computing

Across Europe and the World, quantum computing is gathering investment from states and organizations, as well as private investors. In 2022, the investment in quantum technology was globally around 30 billion euro; in 2023, the investment amounts to 36 billion euro. By 2028, the overall investment in quantum technology is projected to reach globally 53.2 billion euro, and quantum computing investment alone is estimated to be around 17.6 billion euro [27]. Several major players are [28]:

1. **European Union** - The EU Chips Act, with a total budget of around 43 billion euros, has a quantum component included, and the European Quantum Flagship program invests around 1 billion euro in quantum computing, excluding other quantum technologies like quantum sensing.
2. **USA** - The USA Chips Act, with a total budget of around 50 billion euro, has a quantum component included, and the US National Quantum initiative invests around 3.75 billion euro in quantum computing alone.
3. **China** - One of the leading players in quantum computing alongside the USA. Its quantum initiative invests around 15 billion euro in quantum computing.

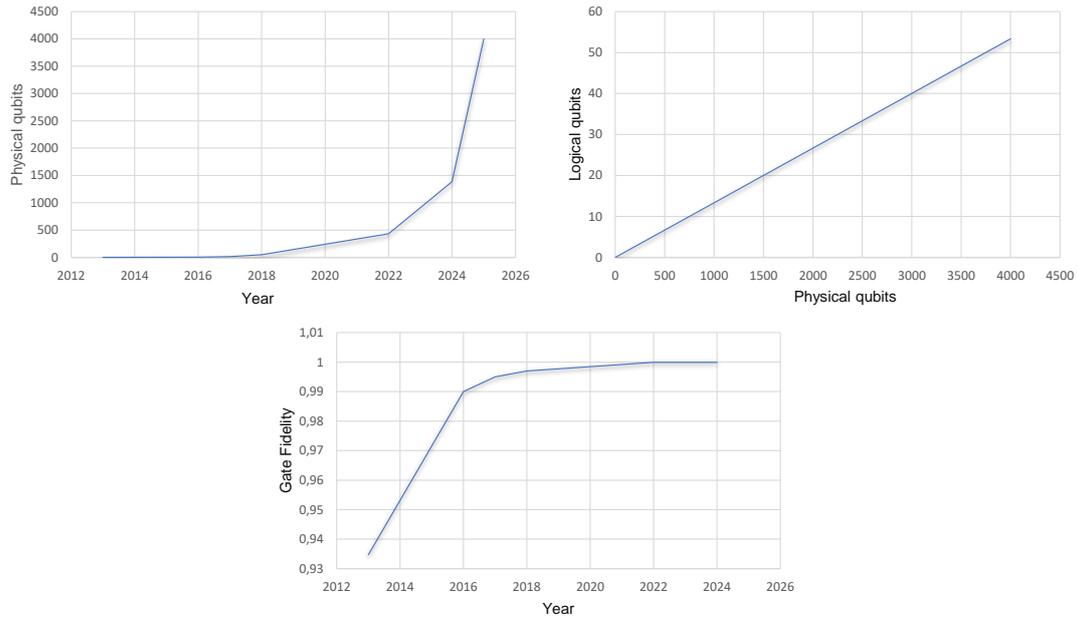


Fig. 5 [Top Left] the trend line for scaling superconducting-based error-prone qubits, or physical qubits, [Top Right] their corresponding logical qubits, when we assume that a single logical qubit is represented by 75 physical qubits based on the error-correction [12], [Bottom] the gate fidelity over the years.

5 When can we expect quantum advantage in climate change detection, climate modeling, and climate digital twin?

The United States is perceived as the leading player in quantum technology, even though Europe has made the most public investments in the industry. In the United States, big technology enterprises such as Microsoft, Google, Intel and IBM has driven commercial development efforts. In Europe, development has been slowed down by fragmentation. Currently, there are about 140 projects, less than half commercial. Many of the groups listed are universities or government labs, or departments within larger tech companies. Here, we can make a distinction between two approaches:

- components provided addressing parts of the HW stack which then may be integrated using so-called open architecture,
- a system integrator capable of bringing together and coordinating all the needed competencies and components that will make up a commercially viable quantum computer.

Superconducting qubit-based approaches are the most researched (and have received the most development resources). Almost all the startups in this space are based on technology from university labs. Manufacture a stable QC requires more than an

exploratory chip. As of early 2023, there have been around a dozen successful attempts to build quantum computers worldwide. There are some specialized companies that are developing middleware for the calibration, management, and optimization of quantum computers to overcome some of the problems caused by errors.

Estimating if and when scalable and useful quantum computers will be available is a difficult art and science. The opinion spread between optimists and pessimists is quite large. As published in their 2020 roadmaps, Google, IBM and Amazon expects to achieve true quantum supremacy relatively quickly and create a quantum computer with 100 logical qubits in less than a decade. On the other end, there are some pessimistic views saying that there is no hope to reach quantum speedup ever. There is not really any strong scientific obstacle preventing the creation of reliable quantum computers. In the scientific community, there is a belief that the uncertainty is mostly a technological and engineering one, and the pace to quantum usability is accelerating. However, there is some pessimism about the ability to fix the noise that affects qubits, whatever their type. A potentially exponential quantum speedup provided by quantum computers may vanish when there is big data that needs to be loaded from classical data or when the full solution vector should be read out [29]. Generally, quantum computers are considered practical for "big compute" problems on small data, not big data problems. There is a growing number of informative end-to-end resource analyses, but typically, these single out very specific algorithms and hardware and make very different assumptions across the stack. Different choices can result in different resource requirements. One can, for example, trade-off more qubits against shorter run times or trade-off faster qubit gate operations against lower fidelities. It is obvious that the number of physical qubits and the duration of a logical time step reduce as physical error rates improve. Entanglement has long been considered to play an essential role in quantum computing and promise for exponential speedup of various quantum algorithms that require asymptotically fewer operations than their classical counterpart. Specific examples where this is the case are quantum problems in chemistry and materials science. Entanglement can be seen as the key feature that sets quantum computing apart from classically simulable processes. Thus, the key metrics to follow the development should include the number and quality of entangling gates provided. The GHZ states provide the strongest non-local correlations for an n-particle entangled state. These GHZ states are very fragile, as the loss of a single particle completely destroys the entanglement. Also, because all particles contribute to phase evolution, the dephasing time decreases with the particle number. Such states are challenging to create, requiring either many particles to interact with each other or a series of two-particle interactions performed in sequence. Some of the recent approaches to improving the SC qubit fidelities include

- redesigning the qubit geometries,
- use of new low-loss materials and
- optimizing the control pulse that drives the quantum system.

Based on the current quantum volume indicator, a marketing simplification tool from IBM, current quantum computers are more than easily emulable in a simple classical computer. In current NISQ computers, the fully manageable number of qubits is

somewhere below 25, way below 50, which is the limit for emulation. Quantum volume sets a limit to Hilbert’s vector space, i.e. the number of different superposed states that are manageable from a practical point of view with a depth of computation equal to the number of corresponding qubits. Based on the expert estimation, we believe that starting in 2025, we will see some relevant quantum advantages with actual data and useful algorithms running on NISQ hardware in climate change detection, climate modeling, and climate digital twin domain. A quantum advantage can come from the computing time, system energetic footprint, and/or the precision of the outcome (metrics: time to solution, energy consumed to reach the solution, and precision of the solution). We estimate that the threshold of 150-ish high-quality qubits, with a low error rate and a long coherence time, will be needed to achieve any real quantum advantage. With these qubits, it may be possible to form about ten logical qubits. However, entangled qubits are required for exponential speedup and significant quantum advantage. We estimate that the number of maximally entangled logical qubits will start growing exponentially around 2030 with advancements in qubit engineering. We summarise this development in three phases.

1. Late NISQ era: (100 – 200+ physical qubits; 99.99%+ fidelities, especially 2Q gate fidelity; high qubit connectivity) (3 – 5 years from now).
2. Early Fault Tolerant QC era delivering significant advantage (< 10 maximally entangled logical qubits) (5 – 10 years from now).
3. Fault Tolerant QC era delivering exponential advantage (> 50 maximally entangled logical qubits) (10 – 20 years from now).

6 Quantum for climate change detection

Earth observation satellites capture changes on Earth’s surface, and the captured signals are in a very narrow spectral band. For example, an Environmental Mapping (EnMAP) satellite detects spectral wavelengths in ranges of 420 nm to 1000 nm and from 900 nm to 2450 nm. Its main task is to collect hyperspectral imaging data in order to provide vital information for climate change detection and environmental monitoring, such as climate change impact and land cover changes [30]. However, current DL techniques and conventional numerical methods for climate change detection and environmental monitoring are costly in terms of computational time and electric power consumption. There are three possible quantum approaches to tackle this problem:

1. The first one is Variational Quantum Algorithms (VQAs): VQAs are a class of Quantum Machine Learning (QML) models aimed at the application in the NISQ era. These algorithms employ jointly parameterized quantum Circuits (PQCs) and classical optimization techniques for finding optimal quantum circuits that have desirable properties from the point of a given application. From the perspective of computational time required and electrical power consumed, VQAs require less training datasets compared to conventional DL models [31] - it implies faster training time than its counterpart classical technique, whereas quantum machines also consume less electric power than supercomputers at the same time [2] (e.g., a D-Wave quantum annealer operates at around 25 kW power, whereas the Summit

Table 3 Summary of the identified feature selection methods for hyperspectral imagery data: RFE-Recursive Feature Elimination, QSVM-Quantum Support Vector Machine, and VQAs-Variational Quantum Algorithms.

Method	RFE for QSVM [47]	RFE for VQAs [48]	Quantum optimization [40]
Resources	high $\gtrsim 10^5$ logical qubits	moderate $\sim 10^2$ logical qubits	low/moderate $\sim 10^2$ logical qubits
Time horizon	> 15 years	3-5 years	now/3-5 years
Architecture	gate-based quantum	gate-based hybrid	annealing/gate-based hybrid
Speedup	exponential	polynomial	polynomial/exponential

supercomputer consumes around 13 MW power). VQAs are already applied to, for example, change detection [32, 33], chlorophyll concentration estimation in water [34], detecting clouds [35], and phase unwrapping for synthetic aperture radar datasets [36, 37].

2. The second approach is feature reduction and selection: Feature selection and feature extraction are common methods for reducing the number of features in large, high-dimensional data sets. A basic distinction between these methods is that the first involves transforming the original features, while the second preserves the features. The procedures have profound practical consequences, allowing for less electric power consumption and more effective data storage. The hyperspectral data satellite data, with even hundreds of narrow spectral bands, provide an example of the area in which utilization of the methods seems virtually unavoidable. The rich spectral information may simply surpass the needs of certain applications. On the other hand, since the number of possible selections (subsets) grows exponentially with the number of features, the application of the selection methods involves hard optimization tasks (see Tables 3 and 4). Another approach is to select the core of a dataset (“coreset”) that is representative of an original dataset [38, 39]. There are already some first attempts for selecting informative features [40] and assembling the coreset from satellite datasets [38, 41]. By either selecting an informative subset feature, reducing the dimensionality, or assembling the coreset of high-dimensional datasets via a quantum approach, the training time and the electric power consumption of both QML and DL models can be reduced substantially.
3. The third approach is to integrate physics laws and models with practical datasets and QML models when a physical model for an event is known, and data is scarce in nature. Here, Quantum Physics-Informed Neural Networks (QPINNs) proposed by the authors of the articles [42, 43] can be applied to, e.g., a rainfall-runoff model that is used for the prediction of flooding and drought analysis [44]. Here, PINNs are ML and DL models imposed by physics laws and PDEs [45, 46], and QPINNs refer to PINNs whose conventional NNs are replaced by QML models. Using QPINNs, we can tackle climate-related challenges and generate better prediction and projection probabilities for about-to-fold as well as already unfolded events than conventional PINNs, when data is too small in quantity for data-driven methods and decision-making time is a critical factor for human-centered decisions [31].

Table 4 Summary of the identified feature extraction methods for hyperspectral imagery data: QPCA-Quantum Principal Component Analysis, and QAutoencoders-Quantum Autoencoders.

Method	QPCA [49]	variational QPCA [50]	QAutoencoders [51]
Resources	high $\gtrsim 10^3$ logical qubits	moderate $\sim 10^2$ logical qubits	low/moderate $\sim 10^1 - 10^2$ logical qubits
Time horizon	15 years	3-5 years	3-5 years
Architecture	gate-based quantum	gate-based hybrid	gate-based hybrid
Speedup	polynomial	polynomial	polynomial

7 Quantum for climate modeling

Climate modeling refers to modeling the behavior of the climate system for predicting and projecting the Earth’s climate [52]. The prediction and projection of climate models depend on the so-called grid cells, each of which represents the point on/in the Earth. The grid cells are characterized by the spatial resolution and their evolution governed by a climate model is defined by the temporal resolution. We note that the amount of data in a climate model is large. With a typical spatial resolution of 10 km, the total number of grid cells representing the atmosphere is in the hundreds of millions. Each grid cell has several variables associated with it, such as air density, temperature, wind speed, humidity, etc. The total parameter space is thus counted in the billions. The finer the spatial and temporal resolution, the more computationally expensive the climate model; climate models governed by PDEs generate better outputs than pure data-driven approaches but are computationally expensive as the spatial and temporal resolution get finer [53]. Doubling the model’s resolution typically requires halving the time steps, following the Courant-Friedrichs-Lewy condition [54]. Thus, doubling the resolution, e.g., going from 10 km to 5 km increases the computational cost roughly by a factor of 8. To tackle computationally expensive climate DL and climate PDEs using quantum algorithms, we could utilize the following approaches:

1. VQAs can be used to test and solve climate PDEs [5] since they have more expressive power than their classical counterparts [55],
2. Due to the limitation of the memory capacity of computing devices and large-scale climate datasets, we need to train conventional DL models on a small subset of climate datasets, and however, they do not generalize well on small-scale datasets compared to large-scale ones [56]. To overcome the small dataset challenge, QPINNs can be utilized for predicting and projecting some climate states [42, 43, 53].
3. Another promising approach is to decrease the spatial resolution of grid cells without losing accuracy by using climate QML models for interpolation identical to a conventional classical method [57],
4. Quantum machines can be used to simulate atmospheric chemistry [58]. Having fast, highly accurate methods for simulating atmospheric chemistry is crucial, as the number of possible reaction pathways also grows rapidly with the size of

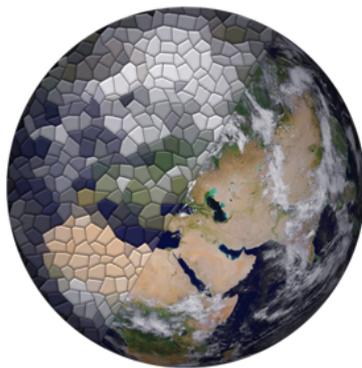


Fig. 6 Digital twins of the Earth attempt to replicate the behavior of certain aspects of the planet based on Earth Observation data and physical models.

the molecules involved in the reactions. Here, quantum chemistry algorithms and quantum machines can play a decisive role.

In addition to quantum approaches for the computational time and electrical power consumption reduction, quantum-inspired algorithms like quantum tensor network-based methods may also help decrease the time and computational cost for tackling climate change detection tasks and climate models [59]. Another advantage of quantum tensor network-based methods is that we can deploy them on an HPC system and quantum machines and utilize them to benchmark the performance of quantum machines with respect to an HPC system [60, 61]. We can also utilize quantum tensor networks for compressing climate Deep Neural Networks (DNNs) and climate PINNs to decrease their computational time and electrical power consumption [59, 62]. The impact of quantum machines will be, therefore, enormous for processing satellite-based datasets and computational methods for climate change detection and climate modeling for making high stake decisions (safety-critical and human-centered decisions) when we have an access to reasonable noisy intermediate-scale and fault-tolerant QCs integrated with an HPC system: HPC+QCs for intractable computational problems of practical significance.

8 Quantum for climate digital twin

8.1 Climate digital twin

The Climate Adaptation Digital Twin (ClimateDT) is a project issued by the European Centre for Medium-Range Weather Forecasts (ECMWF) in the Destination Earth initiative, where the goal is to develop a highly accurate digital model of the Earth (see Fig. 6). The aim is to develop an accurate model of the Earth in order to monitor and simulate the interactions between the natural environment and human activities with as high precision as possible. Through this, the effects of various natural phenomena and human actions on the climate can be studied. The underlying goal

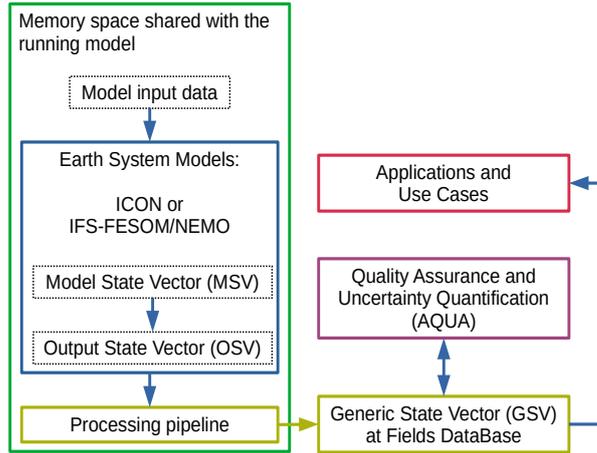


Fig. 7 Operational version of the ClimateDT workflow that will be developed during 2023.

is to move from plausibility assessments of local and regional climate to fully developed risk assessments. The Climate Digital Twin (ClimateDT) workflow is presented in Fig. 7. The workflow begins with the typical initialization and preparatory steps required by a climate or Earth System Model (ESM). In the Climate DT project, the ESMs in use are ICON and IFS. In the workflow, the current model state, illustrated as a Model State Vector (MSV), is propagated forward in time to produce a new state and, simultaneously, the model output or Output State Vector (OSV). This output is streamed (not saved) through a processing pipeline – that introduces additional diagnostic variables and handles interpolation, meta-data conversion and simple operations on the fields – to generate a Generic State Vector (GSV). The GSV is saved directly to Fields DataBase, which is a domain-specific object store developed at the ECMWF; another streaming approach is also being developed with the use of Maestro (<https://www.maestro-data.eu/>). The GSV is then forwarded to the applications and quality assessment and uncertainty quantification (AQUA), all of which can also utilize external data sources, e.g., observations, climatologies, and reanalysis. Indeed, the most resource-heavy and time-consuming part of this workflow, i.e., the bottleneck, is the climate model itself. Fig. 8 shows the relation between different processes in the ICON-Sapphire Earth system model [63]. What can be seen is that different processes are updated at different intervals, that is, with different Δt . This is partly due to the varying computational complexity for propagating specific processes in time in the Earth and climate models. The shortest time steps are those of the dynamical core computations that solve the fluid dynamics equations of atmospheric motions, while the radiative transfer computations have the longest time steps. There is roughly a 1:30 ratio between the shortest and longest time steps. In the latest climate models within ClimateDT, with a resolution of 10 km, the time steps for dynamics and radiation are typically 60 s and 30 min, respectively. Presently, the wall time for computing the individual time steps ranges from the subsecond regime to around 10 s on the LUMI supercomputer.

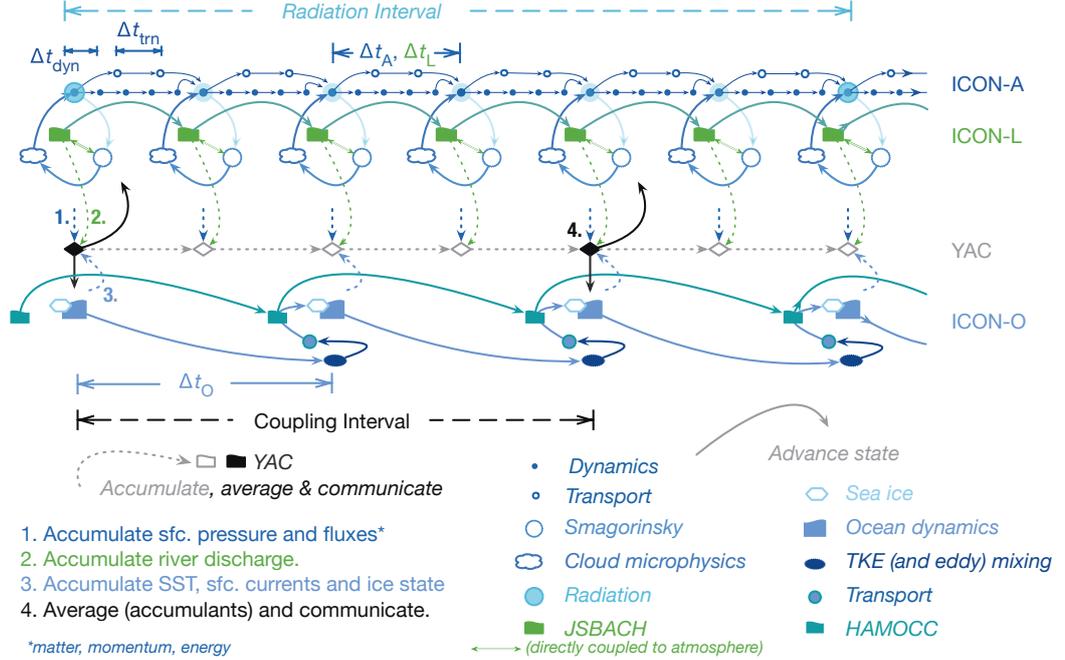


Fig. 8 Time stepping in the ICON-Sapphire Earth system model. ICON-A: atmosphere component; ICON-L: land component; YAC: atmosphere-ocean coupler; ICON-O: ocean component. Reproduced from [63] under the Creative Commons Attribution 4.0 license.

8.2 Missing physics in the climate models

Cloud feedback and cloud-aerosol interactions are the most likely contributors to the high values and increased range of equilibrium climate sensitivity in CMIP6 [64]. In the past, clouds have been poorly represented in Earth System Models (ESMs) due to the complex cloud formation process and because the models could not be run on the scales at which clouds form. Additionally, numerical cloud modeling has relied on the Eulerian continuous medium approach for all cloud thermodynamic variables. However, recently, modeling has shifted towards Lagrangian particle-based probabilistic approaches in small and cloud-scale simulations. Clouds are being taken seriously – the World Climate Research Programme has launched a Grand Challenge on Clouds, Circulation and Climate Sensitivity, and NASA has a Grand Challenge “Uncertainty Project” [65] tackling cloud physics knowledge on ESMs. Clouds are also a focus point for the DYnamics of the Atmospheric general circulation Modeled On Non-hydrostatic Domains (DYAMOND) initiative, where a relatively recent review [66] proposed a protocol for the first intercomparison project of global storm-resolving models. The review presents 40-day global model simulations (these include ICON and IFS) with a grid resolution uniformly lower than 5km and addresses both scientific aspects and computational performance analysis. The outlook is optimistic even though the authors note that fully resolving shallow cloud systems, whose vertical (and hence horizontal)

scale may be only a few kilometers, requires substantially smaller grid distances. This ties in with machine learning efforts for cloud cover modeling [67], and consequently with quantum machine learning efforts discussed above. We expect cloud representation to improve in all ESMs, including ICON and IFS. In the first phase, using purely classical supercomputing and, subsequently, quantum-accelerated HPC.

8.3 Quantum approaches

From the previous section, we can identify two main challenges that hamper the direct adoption of quantum computing to climate modeling problems within ClimateDT:

1. “big data” problem, and
2. “short wall-time” for individual calculations.

First, the climate models work on a large amount of data, both as input and output. However, these “Big data” problems are unsuitable for quantum computers. The strength of quantum computers lies in being able to solve problems with a *moderate* amount of both input and output variables, where the relation between input and output variables is a highly complex equation that can be solved efficiently by some quantum algorithm, exploiting quantum parallelism [68]. In other words, quantum computing typically requires problems with a large potential solution space but only a small set or even a single solution, with the additional provision that the input parameters must be of the same order of magnitude as the number of qubits in the system. Second, for quantum computers to show a wall-time advantage over classical computers, they need to solve sufficiently complex algorithms. This means that the algorithms have to be sufficiently deep; that is, the number of basic operations has to be high. In practice, single useful quantum computing calculations will take at least seconds to complete [69]. Individual variational circuits can and do take a shorter time, but the wall time to solution is, of course, much longer, as several iterations need to be performed. On the other hand, now, the shortest individual time-steps in the climate digital twins take less than a second, and even the longest is around 10 seconds. Further, the aim of the ClimateDT initiative is to speed up the individual time steps significantly, with up to a factor of one hundred. This would push *all* of the individual propagation calculations into the sub-second regime. Thus, quantum computers cannot speed up these calculations further, as they already are faster than the fastest useful quantum computer calculations. Climate models would thus, at first glance, seem to be rather unsuitable for quantum acceleration. To gain some quantum advantage, we need to consider the problem at hand from a broader perspective. Simply taking present classical algorithms and the approximations they include and rely on and transforming these to quantum versions of the same will not work. Instead, the quantum advantage will be found by approaching the problem from different, new angles, utilizing the unique features of quantum machines. A large part of the calculations in the current ClimateDT workflows are, in effect, Computational Fluid Dynamics (CFD or PDEs). Here, we have a direct connection to solving linear systems of equations. The HHL quantum algorithm for linear systems of equations, named after its authors Harrow, Hassidim, and Lloyd [70], and variations thereof, thus have the potential to speed up CFD simulations. As noted by Lapworth [71], classical

algorithms running on supercomputers are highly efficient at solving matrix equations by, for example, side-stepping the need for matrix inversions. Quantum algorithms do not need to, even *should* not rely on the same approximations as classical algorithms, however. Quantum algorithms like HHL and the Quantum Singular Value Transformation (QSVT) [72] can efficiently perform direct matrix inversions and should therefore be utilized for quantum advantage. The approach presented by Lapworth [71] relies on fault-tolerant quantum computers, but hybrid classical/quantum algorithms for the NISQ era have been proposed and discussed [73].

9 Uncertainty quantification for climate change, climate modeling, and climate digital twin

With the mentioned-above quantum solutions, e.g., climate QML models and climate PDEs, the prediction and projection of climate change detection, climate modeling, and ClimateDT (AQUA in the ClimateDT workflow shown in the Fig. 7) using quantum models provide meaningful information with some uncertainty [74]. One approach to quantify the uncertainty of quantum models and to decrease the uncertainty of classical approaches is to integrate Bayesian analysis with quantum models. Quantum models integrated with Bayesian analysis promise to tackle efficiently some hard computational problems on quantum computers [75, 76]. Moreover, they promise to generate solutions to a class of computational problems much faster than conventional computing resources (less time and less electric power usage). Classical Bayesian analysis generating probability distributions of predictions and weights is a natural data-efficient and inherently interpretable model thanks to its respective uncertainties in its predictions and weights [77, 78]. In contrast, conventional DL models and numerical models involving PDEs considered uninterpretable black-box models require big labeled datasets, and they even need to be trained and tested on sub-datasets, including training, test, and validation sets, while one does not need to divide datasets into training, test, and validation sets for Bayesian analysis. For limited labeled datasets, this dataset division raises a challenge for DL and PDEs but not for Bayesian analysis [79]. Moreover, DL and PDEs also yield point estimates of predictions with point weights lacking their uncertainty, or lacking explainability due to the uninterpretable black-box paradigm [80]. DL and PDEs combined with Bayesian analysis are called Probabilistic Numerics (PN) [81], and PN quantifies uncertainties in its predictions and weights since it better utilizes the available dataset, either small or big. Namely, PN models analyze data-driven approaches using Bayesian analysis while their weights and predictions follow certain probability distributions [82]. To design PN for climate change detection and climate modeling via a quantum approach, we first assume a model $F_{\theta} = F_{\theta}(\cdot)$ (a climate QML model or a climate PDE) for a given dataset $\mathcal{S} = \{y_i, \mathbf{x}_i\}_{i=1}^N$. Secondly, its weights and predictions need to be defined according to some prior $p(\theta)$ and likelihood $p(\mathcal{S}|F_{\theta})$ distributions:

$$\begin{aligned} \theta &\sim p(\theta) = \mathcal{N}(0, \sigma^2 \mathbf{I}), \\ p(\mathcal{S}|F_{\theta}) &= p(\mathcal{S}_y|\mathcal{S}_x, F_{\theta}) = \mathcal{N}(\mathcal{S}_y; F_{\theta}(\mathcal{S}_x), \sigma^2 \mathbf{I}); \end{aligned} \tag{1}$$

where weights $\boldsymbol{\theta}$ are sampled from a normal distribution $\mathcal{N}(0, \sigma^2)$ with zero mean and known uncertainty σ^2 . \mathcal{S}_y and \mathcal{S}_x denote labels $\{y_i\}_{i=1}^N$ and input data points $\{\mathbf{x}_i\}_{i=1}^N$ e.g., $F_{\boldsymbol{\theta}}(\mathcal{S}_x)$. We note that one can represent a prior and likelihood by any probability distribution function instead of a normal distribution. For simplicity, we utilized a normal distribution $\mathcal{N}(\cdot)$. To quantify uncertainties in predictions and weights, PN utilizes the Bayes' theorem:

$$p(F_{\boldsymbol{\theta}}|\mathcal{S}) = \frac{p(\mathcal{S}|F_{\boldsymbol{\theta}})p(\boldsymbol{\theta})}{p(\mathcal{S})} \iff p(\boldsymbol{\theta}|\mathcal{S}) = \frac{p(\mathcal{S}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\mathcal{S})}, \quad \text{given } p(\mathcal{S}) = \int_{\Omega_{\boldsymbol{\theta}}} p(\mathcal{S}|\boldsymbol{\theta})p(\boldsymbol{\theta})d\boldsymbol{\theta}; \quad (2)$$

here $p(\boldsymbol{\theta}|\mathcal{S})$ is the posterior, and $p(\mathcal{S})$ is the evidence integrating over parameter space $\Omega_{\boldsymbol{\theta}}$. Finally, after computing the posterior distribution, the expressed by Eq. (2), we can calculate a probability to predict a label \hat{y} given a test data point $\hat{\mathbf{x}}$ and dataset \mathcal{S} , that is, a predictive posterior:

$$p(\hat{y}|\hat{\mathbf{x}}, \mathcal{S}) = \int_{\Omega_{\boldsymbol{\theta}}} p(\hat{y}|\hat{\mathbf{x}}, \boldsymbol{\theta})p(\boldsymbol{\theta}|\mathcal{S})d\boldsymbol{\theta}. \quad (3)$$

The posterior $p(\boldsymbol{\theta}|\mathcal{S})$ gives uncertainties in weights -. This uncertainty is called an epistemic uncertainty, while the predictive likelihood $p(\hat{y}|\hat{\mathbf{x}}, \boldsymbol{\theta})$ yields uncertainties in predictions - this uncertainty is called an aleatoric uncertainty. Therefore, the predictive posterior $p(\hat{y}|\hat{\mathbf{x}}, \mathcal{S})$ generates total uncertainties in predictions by leveraging both epistemic and aleatoric uncertainties [83, 84]. By convention, the epistemic uncertainty related to the random noise (randomness) in a dataset can be reduced by increasing the size of a dataset, while the aleatoric uncertainty associated with a lack of knowledge in a model $\boldsymbol{\theta}$ is an irreducible uncertainty even by increasing the size of a dataset. The parameter space $\Omega_{\boldsymbol{\theta}}$ of a given model includes several thousands to millions of tuneable weights $\boldsymbol{\theta}$. This high dimensional space of weights raises a challenge to integrate the evidence $p(\mathcal{S})$ as well as predictive posterior $p(\hat{y}|\hat{\mathbf{x}}, \mathcal{S})$ over $\Omega_{\boldsymbol{\theta}}$; computing the evidence and predictive posterior is an intractable problem [1]. Hence, the posterior $p(\boldsymbol{\theta}|\mathcal{S})$ is a hard-to-compute function on conventional computers due to the intractable evidence. To tackle these intractability challenges for climate change detection and climate modeling, there exist already some quantum approaches such as quantum Monte Carlo Markov Chain (MCMC) and quantum variational inference (VI) [75, 85]. In contrast to the conventional classical MCMC and VI, their quantum approaches help generate faster and better results for climate QML models and climate PDEs. More importantly, quantum approaches for classical MCMC and VI promise to reduce the uncertainties in conventional climate models due to their better approximation property of a distribution function. Thus, it is of great importance to design and use quantum VI and the quantum MCMC to make them better on approximate samples - reducing the uncertainties in classical change detection methods and climate models - to predict climate change detection and project a climate state.

10 Combining high-performance computing and quantum computing: HPC+QC

There are presently major ongoing efforts around the globe to connect HPC infrastructure with quantum computers. This is perhaps even somewhat surprising, considering that quantum computers presently cannot solve any useful real-world modeling problem more efficiently than a single node of a supercomputer. At the same time, it is a testament to the potential and the *belief* in the potential of quantum computing for scientific modeling. In Europe, the plans for making quantum computing relevant for research and development in academia and industry alike have been outlined, with the goal of having a European quantum computing infrastructure exhibiting quantum advantage by 2030. The first quantum simulators are already being integrated with HPC infrastructure in the HPCQS project [<https://www.hpcqs.eu/>]. In June 2023, the EuroHPC Joint Undertaking signed hosting agreements for six different quantum computers to be placed in HPC centers around Europe, with the plan to make these available to European users in 2024. These first quantum computers are only the beginning; several updates and new procurements are already planned. The actual (future) HPC infrastructure and its implementation must be accounted for. Already in the near-term, it is expected that individual supercomputers will be connected to several quantum machines of various types and implementations [86]. The initial setups, with individual QPUs distributed throughout the continent, connected to an HPC system, can be seen as precursors to a future where QPUs will be connected in parallel, either entangled or not. Plans for even tighter, on-chip integration of QPUs with classical processing units already exist and may be the way to reach fault-tolerant quantum computing. With this in mind, more emphasis on developing parallel quantum algorithms, which simultaneously utilize several QPUs in an HPC+ n QC manner, would seem appropriate. For time-evolution problems like climate modeling, this can be a necessary development at a relatively early stage to enable the quantum processing part to keep up with the classical computing tasks at each time step. Reassuringly, the importance of investing in software development for hybrid HPC+QC applications has been recognized. These developments complement the efforts for developing purely classical software for exascale supercomputers and beyond, exemplified by the Destination Earth initiative. Here, it is apt to note that there is a need for significant classical software development alongside quantum algorithm research. Presently, pre- and post-processing tasks take up a significant portion of the total wall time of executing a quantum algorithm. As an example, in the recent experiment on spin dynamics using IBM's 127 qubits QPU, the actual time spent on the QPU was 5 minutes, while the wall-time of the experiment was a hundred times longer, over 9 hours [87]. These overheads will decrease in the future, but at the same time, increasing the qubit count will again increase the complexity of pre- and post-processing. Part of this overhead lies within the domain of hardware development, e.g., qubit reset and readout. Much of this is, however, classical computing routines, such as compiling, transpiling, qubit routing optimization, error mitigation, and noise canceling, to name a few. All of these will become computationally more demanding with increasing qubit count and will, therefore, require increasing amounts of classical computing

power. Thus, efficiently operating the quantum machines of the future will require an HPC infrastructure and the classical software to run on it. For reaching quantum advantage as soon as possible, both in general and especially within climate modeling, it is important to develop quantum algorithms keeping the immense, existing classical supercomputing power in mind. This means, for example, taking full advantage of the available HPC infrastructure for performing the necessary pre-and post-processing of data to and from the quantum machines. For electronic structure problems, as in the case of modeling atmospheric reactions discussed above, HPC resources are needed for providing an initial guess for the quantum computer; in other words, they provide the best approximation to the true electronic structure that classical methods can provide, and refine it further on the quantum computer. This exemplifies the need for a broad, multidisciplinary approach to quantum advantage. We need to combine expertise in quantum algorithms, classical HPC algorithms, computer science, AI/ML, and specific domain expertise, also from the end-user side.

11 SWOT analysis

11.1 Strengths

- Quantum machines could be applied to generate data samples from classically difficult distributions [88].
- Proved exponential speed-up in at least one scenario [89].
- The climate modeling community deeply understands the problem at hand and the bottlenecks present, both from the efficiency and accuracy points of view.
- A recognized high-priority problem: resources available for finding solutions.

11.2 Weaknesses

- Data loading is a major obstacle for achieving exponential speed-up of some QML algorithms [29].
- Measurement error mitigation is strongly limited by the number of qubits and the circuit depth. [90].
- Quantum machines can be difficult to train due to the error correction scheme [91].
- Understanding of the applicability of quantum computing to climate modeling limited.
- Quantum-acceleration is presently not seen as a viable route due to the “big data” nature of digital twins.

11.3 Opportunities

- Major shift in the quality of quantum computers. NISQ machines may be available with less than 100 high-quality error-prone qubits.
- New applications of classical machine learning for quantum computing: compiling, mapping, control, error correction.
- Potential to utilize hybrid approaches that require a relatively small number of qubits (of the order on 10^2 logical qubits), thereby increasing feasibility.
- Progress in QC hardware and software capacity can enable more accurate models.

- Global drive for supporting hybrid HPC+QC software development.

11.4 Threats

- Fundamental lack of ability to control, mitigate, and correct sources of noise in the quantum machines.
- Novel classical algorithms inspired by quantum computing may outperform some pure quantum algorithms.
- Development of sufficiently powerful QC hardware/software delayed.
- Lack of long-term funding commitment to development, in case near-term gains do not live up to (inflated) expectations.

12 Conclusion

Quantum machines promise to solve a certain class of computational problems faster than conventional machines. In particular, the hardness of the computational problems can be measured from the perspective of the computational complexity theory. Hence, this study identifies intractable climate problems that can not be efficiently solved on classical supercomputers, but quantum machines promise to find their solutions faster and more energy efficient than their classical counterparts. In addition, we assess and examine distinct quantum machines, including a quantum annealer, a quantum simulator, and universal quantum computers, for their practicality. Toward practical problems, we propose climate change detection, climate modeling, and climate digital twin use-case instances. In particular, we analyzed and evaluated the hardness of our practical climate challenges based on the computational complexity theory and the computational time and energy consumption required.

Acknowledgment

This project article dubbed “Quantum Advantage for Earth Observation (QA4EO)” is funded by the European Space Agency (ESA) [92].

References

- [1] Arora, S., Barak, B.: Computational complexity: a modern approach. Cambridge University Press (2009)
- [2] Berger, C., Paolo, A.D., Forrest, T., Hadfield, S., Sawaya, N., Stechly, M., Thibault, K.: Quantum technologies for climate change: Preliminary assessment (2021) [arXiv:2107.05362](https://arxiv.org/abs/2107.05362) [quant-ph]
- [3] Stollenwerk, T., Hadfield, S., Wang, Z.: Toward quantum gate-model heuristics for real-world planning problems. *IEEE Transactions on Quantum Engineering* **1**, 1–16 (2020) <https://doi.org/10.1109/TQE.2020.3030609>
- [4] Rainjonneau, S., Tokarev, I., Iudin, S., Rayaprolu, S., Pinto, K., Lemtiuzhnikova, D., Koblan, M., Barashov, E., Kordzanganeh, M., Pflitsch, M., Melnikov, A.:

- Quantum algorithms applied to satellite mission planning for earth observation. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **16**, 7062–7075 (2023) <https://doi.org/10.1109/JSTARS.2023.3287154>
- [5] Demirdjian, R., Gunlycke, D., Reynolds, C.A., Doyle, J.D., Tafur, S.: Variational quantum solutions to the advection–diffusion equation for applications in fluid dynamics. *Quantum Information Processing* **21**(9), 322 (2022) <https://doi.org/10.1007/s11128-022-03667-7>
- [6] Paudel, H.P., Syamlal, M., Crawford, S.E., Lee, Y.-L., Shugayev, R.A., Lu, P., Ohodnicki, P.R., Mollot, D., Duan, Y.: Quantum computing and simulations for energy applications: Review and perspective. *ACS Engineering Au* **2**(3), 151–196 (2022) <https://doi.org/10.1021/acsengineeringau.1c00033>
- [7] Giani, A.: Quantum computing opportunities in renewable energy. *Nature Computational Science* **1**(2), 90–91 (2021) <https://doi.org/10.1038/s43588-021-00032-z>
- [8] Otgonbaatar, S., Kranzlmüller, D.: Exploiting the Quantum Advantage for Satellite Image Processing: Quantum Resource Estimation (2023) <https://doi.org/10.36227/techrxiv.24085122.v1>
- [9] Otgonbaatar, S., Schwarz, G., Dacu, M., Kranzlmüller, D.: Quantum transfer learning for real-world, small, and high-dimensional remotely sensed datasets. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 1–8 (2023) <https://doi.org/10.1109/JSTARS.2023.3316306>
- [10] Beverland, M.E., Murali, P., Troyer, M., Svore, K.M., Hoefler, T., Kliuchnikov, V., Low, G.H., Soeken, M., Sundaram, A., Vaschillo, A.: Assessing requirements to scale to practical quantum advantage (2022) [arXiv:2211.07629](https://arxiv.org/abs/2211.07629) [quant-ph]
- [11] European Quantum Industry Consortium (QuIC). <https://www.euroquic.org/>
- [12] Fowler, A.G., Gidney, C.: Low overhead quantum computation using lattice surgery (2019) [arXiv:1808.06709](https://arxiv.org/abs/1808.06709) [quant-ph]
- [13] Cheng, B., Deng, X.-H., Gu, X., He, Y., Hu, G., Huang, P., Li, J., Lin, B.-C., Lu, D., Lu, Y., Qiu, C., Wang, H., Xin, T., Yu, S., Yung, M.-H., Zeng, J., Zhang, S., Zhong, Y., Peng, X., Nori, F., Yu, D.: Noisy intermediate-scale quantum computers. *Frontiers of Physics* **18**(2) (2023) <https://doi.org/10.1007/s11467-022-1249-z>
- [14] Cheng, B., Deng, X.-H., Gu, X., He, Y., Hu, G., Huang, P., Li, J., Lin, B.-C., Lu, D., Lu, Y., Qiu, C., Wang, H., Xin, T., Yu, S., Yung, M.-H., Zeng, J., Zhang, S., Zhong, Y., Peng, X., Nori, F., Yu, D.: Noisy intermediate-scale quantum computers. *Frontiers of Physics* **18**(2), 21308 (2023) <https://doi.org/10.1007/s11467-022-1249-z>

- [15] DiVincenzo, D.P.: The physical implementation of quantum computation. *Fortschritte der Physik: Progress of Physics* **48**(9-11), 771–783 (2000)
- [16] Quantum Roadmap. <https://databaseline.tech/quantum.html>
- [17] Acharya, R., Aleiner, I., Allen, R., Andersen, T.I., Ansmann, M., Arute, F., Arya, K., Asfaw, A., Atalaya, J., Babbush, R., Bacon, D., Bardin, J.C., Basso, J., Bengtsson, A., Boixo, S., Bortoli, G., Bourassa, A., Bovaird, J., Brill, L., Broughton, M., Buckley, B.B., Buell, D.A., Burger, T., Burkett, B., Bushnell, N., Chen, Y., Chen, Z., Chiaro, B., Cogan, J., Collins, R., Conner, P., Courtney, W., Crook, A.L., Curtin, B., Debroy, D.M., Del Toro Barba, A., Demura, S., Dunsworth, A., Eppens, D., Erickson, C., Faoro, L., Farhi, E., Fatemi, R., Flores Burgos, L., Forati, E., Fowler, A.G., Foxen, B., Giang, W., Gidney, C., Gilboa, D., Giustina, M., Grajales Dau, A., Gross, J.A., Habegger, S., Hamilton, M.C., Harrigan, M.P., Harrington, S.D., Higgott, O., Hilton, J., Hoffmann, M., Hong, S., Huang, T., Huff, A., Huggins, W.J., Ioffe, L.B., Isakov, S.V., Iveland, J., Jeffrey, E., Jiang, Z., Jones, C., Juhas, P., Kafri, D., Kechedzhi, K., Kelly, J., Khattar, T., Khezri, M., Kieferová, M., Kim, S., Kitaev, A., Klimov, P.V., Klots, A.R., Korotkov, A.N., Kostrița, F., Kreikebaum, J.M., Landhuis, D., Laptev, P., Lau, K.-M., Laws, L., Lee, J., Lee, K., Lester, B.J., Lill, A., Liu, W., Locharlar, A., Lucero, E., Malone, F.D., Marshall, J., Martin, O., McClean, J.R., McCourt, T., McEwen, M., Megrant, A., Meurer Costa, B., Mi, X., Miao, K.C., Mohseni, M., Montazeri, S., Morvan, A., Mount, E., Mruzckiewicz, W., Naaman, O., Neeley, M., Neill, C., Nersisyan, A., Neven, H., Newman, M., Ng, J.H., Nguyen, A., Nguyen, M., Niu, M.Y., O’Brien, T.E., Opremcak, A., Platt, J., Petukhov, A., Potter, R., Pryadko, L.P., Quintana, C., Roushan, P., Rubin, N.C., Saei, N., Sank, D., Sankaragomathi, K., Satzinger, K.J., Schurkus, H.F., Schuster, C., Shearn, M.J., Shorter, A., Shvarts, V., Skrzny, J., Smelyanskiy, V., Smith, W.C., Sterling, G., Strain, D., Szalay, M., Torres, A., Vidal, G., Villalonga, B., Vollgraff Heidweiller, C., White, T., Xing, C., Yao, Z.J., Yeh, P., Yoo, J., Young, G., Zalcman, A., Zhang, Y., Zhu, N., AI, G.Q.: Suppressing quantum errors by scaling a surface code logical qubit. *Nature* **614**(7949), 676–681 (2023) <https://doi.org/10.1038/s41586-022-05434-1>
- [18] Pogorelov, I., Feldker, T., Marciniak, C.D., Postler, L., Jacob, G., Krieglsteiner, O., Podlesnic, V., Meth, M., Negnevitsky, V., Stadler, M., Höfer, B., Wächter, C., Lakhmanskiy, K., Blatt, R., Schindler, P., Monz, T.: Compact ion-trap quantum computing demonstrator. *PRX Quantum* **2**, 020343 (2021) <https://doi.org/10.1103/PRXQuantum.2.020343>
- [19] Henriët, L., Béguin, L., Signoles, A., Lahaye, T., Browaeys, A., Reymond, G.-O., Jurczak, C.: Quantum computing with neutral atoms. *Quantum* **4**, 327 (2020) <https://doi.org/10.22331/q-2020-09-21-327>
- [20] Bartolucci, S., Birchall, P., Bombín, H., Cable, H., Dawson, C., Gimeno-Segovia, M., Johnston, E., Kieling, K., Nickerson, N., Pant, M., Pastawski, F., Rudolph,

- T., Sparrow, C.: Fusion-based quantum computation. *Nature Communications* **14**(1), 912 (2023) <https://doi.org/10.1038/s41467-023-36493-1>
- [21] Stano, P., Loss, D.: Review of performance metrics of spin qubits in gated semiconducting nanostructures. *Nature Reviews Physics* **4**(10), 672–688 (2022) <https://doi.org/10.1038/s42254-022-00484-w>
- [22] Liao, H., Wang, D.S., Sitdikov, I., Salcedo, C., Seif, A., Mineev, Z.K.: Machine learning for practical quantum error mitigation (2023) [arXiv:2309.17368](https://arxiv.org/abs/2309.17368) [quant-ph]
- [23] Quantum Edge. <https://quantrolox.com/>
- [24] Gidney, C., Fowler, A.G.: Efficient magic state factories with a catalyzed $|CCZ\rangle$ to $2|T\rangle$ transformation. *Quantum* **3**, 135 (2019) <https://doi.org/10.22331/q-2019-04-30-135>
- [25] Litinski, D.: A game of surface codes: Large-scale quantum computing with lattice surgery. *Quantum* **3**, 128 (2019) <https://doi.org/10.22331/q-2019-03-05-128>
- [26] Reiher, M., Wiebe, N., Svore, K.M., Wecker, D., Troyer, M.: Elucidating reaction mechanisms on quantum computers. *Proceedings of the National Academy of Sciences* **114**(29), 7555–7560 (2017) <https://doi.org/10.1073/pnas.1619152114> <https://www.pnas.org/doi/pdf/10.1073/pnas.1619152114>
- [27] An Action Plan for a Quantum Technology. https://www.researchandmarkets.com/reports/5317365/quantum-technology-market-by-computing?utm_source=GNOM&utm_medium=PressRelease&utm_code=xt7dw4&utm_campaign=1668311+-+Global+Quantum+Technology+Market+Research+Report+2022-2027%3a+Quantum+Dots+Market+will+Reach+%2413.25+Billion+by+2027%2c+Growing+a+25.1%25+CAGR+and+Led+by+Displays&utm_exec=chdo54prd
- [28] An Action Plan for a Universal Quantum Computer. <https://thequantuminsider.com/2023/05/03/germany-announces-3-billion-euro-action-plan-for-a-universal-quantum-computer/>
- [29] Tang, E.: Quantum Principal Component Analysis Only Achieves an Exponential Speedup Because of Its State Preparation Assumptions. *Physical Review Letters* **127**(6), 060503 (2021) <https://doi.org/10.1103/PhysRevLett.127.060503>. Accessed 2021-08-05
- [30] German Hyperspectral Satellite: EnMAP. <https://www.enmap.org/mission/>
- [31] Caro, M.C., Huang, H.-Y., Cerezo, M., Sharma, K., Sornborger, A., Cincio,

- L., Coles, P.J.: Generalization in quantum machine learning from few training data. *Nature Communications* **13**(1), 4919 (2022) <https://doi.org/10.1038/s41467-022-32550-3>
- [32] Otgonbaatar, S., Datcu, M.: Classification of remote sensing images with parameterized quantum gates. *IEEE Geoscience and Remote Sensing Letters* **19**, 1–5 (2022) <https://doi.org/10.1109/LGRS.2021.3108014>
- [33] Gawron, P., Lewiński, S.: Multi-spectral image classification with quantum neural network. In: *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*, pp. 3513–3516 (2020). IEEE
- [34] Pasetto, E., Riedel, M., Melgani, F., Michielsen, K., Cavallaro, G.: Quantum svr for chlorophyll concentration estimation in water with remote sensing. *IEEE Geoscience and Remote Sensing Letters* **19**, 1–5 (2022) <https://doi.org/10.1109/LGRS.2022.3200325>
- [35] Miroszewski, A., Mielczarek, J., Czelusta, G., Szczepanek, F., Grabowski, B., Le Saux, B., Nalepa, J.: Detecting clouds in multispectral satellite images using quantum-kernel support vector machines. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **16**, 7601–7613 (2023) <https://doi.org/10.1109/JSTARS.2023.3304122>
- [36] Glatting, K., Meyer, J., Huber, S., Krieger, G.: Quantum optimization and quantum machine learning for phase unwrapping in sar interferometry. In: *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)* (2023). <https://elib.dlr.de/195031/>
- [37] Otgonbaatar, S., Datcu, M.: Quantum annealer for network flow minimization in insar images. In: *EUSAR 2021; 13th European Conference on Synthetic Aperture Radar*, pp. 1–4 (2021)
- [38] Otgonbaatar, S., Datcu, M.: Assembly of a coresets of earth observation images on a small quantum computer. *Electronics* **10**(20) (2021) <https://doi.org/10.3390/electronics10202482>
- [39] Tomesh, T., Gokhale, P., Anschuetz, E.R., Chong, F.T.: Coresets clustering on small quantum computers. *Electronics* **10**(14) (2021) <https://doi.org/10.3390/electronics10141690>
- [40] Otgonbaatar, S., Datcu, M.: A quantum annealer for subset feature selection and the classification of hyperspectral images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **14**, 7057–7065 (2021) <https://doi.org/10.1109/JSTARS.2021.3095377>
- [41] Otgonbaatar, S., Datcu, M., Demir, B.: Coresets of hyperspectral images on a small quantum computer. In: *IGARSS 2022 - 2022 IEEE International Geoscience*

- and Remote Sensing Symposium, pp. 4923–4926 (2022). <https://doi.org/10.1109/IGARSS46834.2022.9884273>
- [42] Vadyala, S.R., Betgeri, S.N.: General implementation of quantum physics-informed neural networks. *Array* **18**, 100287 (2023) <https://doi.org/10.1016/j.array.2023.100287>
- [43] Markidis, S.: On physics-informed neural networks for quantum computers. *Frontiers in Applied Mathematics and Statistics* **8** (2022) <https://doi.org/10.3389/fams.2022.1036711>
- [44] Xie, K., Liu, P., Zhang, J., Han, D., Wang, G., Shen, C.: Physics-guided deep learning for rainfall-runoff modeling by considering extreme events and monotonic relationships. *Journal of Hydrology* **603**, 127043 (2021) <https://doi.org/10.1016/j.jhydrol.2021.127043>
- [45] Sedykh, A., Podapaka, M., Sagingalieva, A., Smertyak, N., Pinto, K., Pflitsch, M., Melnikov, A.: Quantum physics-informed neural networks for simulating computational fluid dynamics in complex shapes. arXiv e-prints, 2304–11247 (2023) <https://doi.org/10.48550/arXiv.2304.11247> arXiv:2304.11247 [cs.LG]
- [46] Raissi, M., Perdikaris, P., Karniadakis, G.E.: Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics* **378**, 686–707 (2019) <https://doi.org/10.1016/j.jcp.2018.10.045>
- [47] Rebentrost, P., Mohseni, M., Lloyd, S.: Quantum support vector machine for big data classification. *Phys. Rev. Lett.* **113**, 130503 (2014) <https://doi.org/10.1103/PhysRevLett.113.130503>
- [48] Havlíček, V., Córcoles, A.D., Temme, K., Harrow, A.W., Kandala, A., Chow, J.M., Gambetta, J.M.: Supervised learning with quantum-enhanced feature spaces. *Nature* **567**(7747), 209–212 (2019) <https://doi.org/10.1038/s41586-019-0980-2>
- [49] Lloyd, S., Mohseni, M., Rebentrost, P.: Quantum principal component analysis. *Nature Physics* **10**(9), 631–633 (2014)
- [50] Gordon, M.H., Cerezo, M., Cincio, L., Coles, P.J.: Covariance matrix preparation for quantum principal component analysis. *PRX Quantum* **3**(3), 030334 (2022)
- [51] Romero, J., Olson, J.P., Aspuru-Guzik, A.: Quantum autoencoders for efficient compression of quantum data. *Quantum Science and Technology* **2**(4), 045001 (2017)
- [52] Gettelman, A., Rood, R.B.: In: Gettelman, A., Rood, R.B. (eds.) *Components of the Climate System*, pp. 13–22. Springer, Berlin, Heidelberg (2016). https://doi.org/10.1007/978-3-642-31723-1_2

[//doi.org/10.1007/978-3-662-48959-8_2](https://doi.org/10.1007/978-3-662-48959-8_2)

- [53] Gelbrecht, M., Boers, N., Kurths, J.: Neural partial differential equations for chaotic systems. *New Journal of Physics* **23**(4), 043005 (2021) <https://doi.org/10.1088/1367-2630/abeb90>
- [54] Courant, R., Friedrichs, K., Lewy, H.: Über die partiellen Differenzgleichungen der mathematischen Physik. *Mathematische Annalen* **100**(1), 32–74 (1928) <https://doi.org/10.1007/bf01448839>
- [55] Abbas, A., Sutter, D., Zoufal, C., Lucchi, A., Figalli, A., Woerner, S.: The power of quantum neural networks. *Nature Computational Science* **1**(6), 403–409 (2021) <https://doi.org/10.1038/s43588-021-00084-1>
- [56] Otgonbaatar, S., Schwarz, G., Datcu, M., Kranzlmüller, D.: Quantum transfer learning for real-world, small, and high-dimensional remotely sensed datasets. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 1–8 (2023) <https://doi.org/10.1109/JSTARS.2023.3316306>
- [57] Kochkov, D., Smith, J.A., Alieva, A., Wang, Q., Brenner, M.P., Hoyer, S.: Machine learning-accelerated computational fluid dynamics. *Proceedings of the National Academy of Sciences* **118**(21), 2101784118 (2021) <https://doi.org/10.1073/pnas.2101784118> <https://www.pnas.org/doi/pdf/10.1073/pnas.2101784118>
- [58] Cao, Y., Romero, J., Olson, J.P., Degroote, M., Johnson, P.D., Kieferová, M., Kivlichan, I.D., Menke, T., Peropadre, B., Sawaya, N.P.D., Sim, S., Veis, L., Aspuru-Guzik, A.: Quantum chemistry in the age of quantum computing. *Chemical Reviews* **119**(19), 10856–10915 (2019) <https://doi.org/10.1021/acs.chemrev.8b00803>
- [59] Otgonbaatar, S., Kranzlmüller, D.: Quantum-inspired tensor network for earth science (2023) [arXiv:2301.07528](https://arxiv.org/abs/2301.07528) [physics.geo-ph]
- [60] Huang, H., Liu, X.-Y., Tong, W., Zhang, T., Walid, A., Wang, X.: High performance hierarchical tucker tensor learning using gpu tensor cores. *IEEE Transactions on Computers*, 1–1 (2022) <https://doi.org/10.1109/TC.2022.3172895>
- [61] Huggins, W., Patil, P., Mitchell, B., Whaley, K.B., Stoudenmire, E.M.: Towards quantum machine learning with tensor networks. *Quantum Science and Technology* **4**(2), 024001 (2019) <https://doi.org/10.1088/2058-9565/aaea94>
- [62] Gao, Z.-F., Cheng, S., He, R.-Q., Xie, Z.Y., Zhao, H.-H., Lu, Z.-Y., Xiang, T.: Compressing deep neural networks by matrix product operators. *Phys. Rev. Res.* **2**, 023300 (2020) <https://doi.org/10.1103/PhysRevResearch.2.023300>
- [63] Hohenegger, C., Korn, P., Linardakis, L., Redler, R., Schnur, R., Adamidis, P.,

- Bao, J., Bastin, S., Behraves, M., Bergemann, M., Biercamp, J., Bockelmann, H., Brokopf, R., Brüggemann, N., Casaroli, L., Chegini, F., Datsaris, G., Esch, M., George, G., Giorgetta, M., Gutjahr, O., Haak, H., Hanke, M., Ilyina, T., Jahns, T., Jungclaus, J., Kern, M., Klocke, D., Kluft, L., Kölling, T., Kornblueh, L., Kosukhin, S., Kroll, C., Lee, J., Mauritsen, T., Mehlmann, C., Mieslinger, T., Naumann, A.K., Paccini, L., Peinado, A., Praturi, D.S., Putrasahan, D., Rast, S., Riddick, T., Roeber, N., Schmidt, H., Schulzweida, U., Schütte, F., Segura, H., Shevchenko, R., Singh, V., Specht, M., Stephan, C.C., Storch, J.-S., Vogel, R., Wengel, C., Winkler, M., Ziemann, F., Marotzke, J., Stevens, B.: ICON-Sapphire: simulating the components of the Earth system and their interactions at kilometer and subkilometer scales. *Geoscientific Model Development* **16**(2), 779–811 (2023) <https://doi.org/10.5194/gmd-16-779-2023>
- [64] Meehl, G.A., Senior, C.A., Eyring, V., Flato, G., Lamarque, J.-F., Stouffer, R.J., Taylor, K.E., Schlund, M.: Context for interpreting equilibrium climate sensitivity and transient climate response from the CMIP6 Earth system models. *Science Advances* **6**(26), 1981 (2020) <https://doi.org/10.1126/sciadv.aba1981> <https://www.science.org/doi/pdf/10.1126/sciadv.aba1981>
- [65] Fridlind, A., Lier-Walqui, M., Elsaesser, G., Kelley, M., Ackerman, A., Cesna, G., Schmidt, G.: A grand challenge "uncertainty project" to accelerate advances in earth system predictability: AI-enabled concepts and applications. Technical report, United States (2021). <https://doi.org/10.2172/1769643> . Contract Number: 35604.1. <https://doi.org/10.2172/1769643>
- [66] Stevens, B., Satoh, M., Auger, L., al.: DYAMOND: the DYnamics of the atmospheric general circulation modeled on non-hydrostatic domains. *Progress in Earth and Planetary Science* **6** (2019) <https://doi.org/10.1186/s40645-019-0304-z>
- [67] Grundner, A., Beucler, T., Gentine, P., Iglesias-Suarez, F., Giorgetta, M.A., Eyring, V.: Deep learning based cloud cover parameterization for ICON. *Journal of Advances in Modeling Earth Systems* **14**(12) (2022) <https://doi.org/10.1029/2021ms002959>
- [68] Hoefler, T., Häner, T., Troyer, M.: Disentangling hype from practicality: On realistically achieving quantum advantage. *Commun. ACM* **66**, 82–87 (2023)
- [69] Humble, T.S., McCaskey, A., Lyakh, D.I., Gowrishankar, M., Frisch, A., Monz, T.: Quantum computers for high-performance computing. *IEEE Micro* **41**, 15–23 (2021)
- [70] Harrow, A.W., Hassidim, A., Lloyd, S.: Quantum algorithm for linear systems of equations. *Physical Review Letters* **103**(15) (2009) <https://doi.org/10.1103/physrevlett.103.150502>
- [71] Lapworth, L.: Implicit hybrid quantum-classical CFD calculations using the HHL

- algorithm (2022) [arXiv:2209.07964](https://arxiv.org/abs/2209.07964) [quant-ph]
- [72] Gilyén, A., Su, Y., Low, G.H., Wiebe, N.: Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In: Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing. ACM, ??? (2019). <https://doi.org/10.1145/3313276.3316366> . <https://arxiv.org/abs/1806.01838>
- [73] Kyriienko, O., Paine, A.E., Elfving, V.E.: Solving nonlinear differential equations with differentiable quantum circuits. *Phys. Rev. A* **103**, 052416 (2021) <https://doi.org/10.1103/PhysRevA.103.052416>
- [74] Gettelman, A., Rood, R.B.: Key Concepts in Climate Modeling, pp. 3–12. Springer, Berlin, Heidelberg (2016). https://doi.org/10.1007/978-3-662-48959-8_1 . https://doi.org/10.1007/978-3-662-48959-8_1
- [75] Benedetti, M., Coyle, B., Fiorentini, M., Lubasch, M., Rosenkranz, M.: Variational inference with a quantum computer. *Phys. Rev. Appl.* **16**, 044057 (2021) <https://doi.org/10.1103/PhysRevApplied.16.044057>
- [76] Otgonbaatar, S., Kranzlmüller, D.: Uncertainty quantification: Classical and quantum approaches for limited labelled-datasets. In: Toward Quantum Advantage in High Energy Physics, Max Plank Institute of Quantum Optics (MPQ) (2023). <https://elib.dlr.de/194803/>
- [77] Jospin, L.V., Laga, H., Boussaid, F., Buntine, W., Bennamoun, M.: Hands-on bayesian neural networks—a tutorial for deep learning users. *IEEE Computational Intelligence Magazine* **17**(2), 29–48 (2022) <https://doi.org/10.1109/mci.2022.3155327>
- [78] Koller, C., Kauermann, G., Zhu, X.X.: Going beyond one-hot encoding in classification: Can human uncertainty improve model performance? (2022) [arXiv:2205.15265](https://arxiv.org/abs/2205.15265) [cs.LG]
- [79] Olivier, A., Shields, M.D., Graham-Brady, L.: Bayesian neural networks for uncertainty quantification in data-driven materials modeling. *Computer Methods in Applied Mechanics and Engineering* **386**, 114079 (2021) <https://doi.org/10.1016/j.cma.2021.114079>
- [80] Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead (2018) <https://doi.org/10.48550/ARXIV.1811.10154>
- [81] Hennig, P., Osborne, M.A., Kersting, H.P.: Probabilistic Numerics: Computation as Machine Learning. Cambridge University Press (2022)
- [82] Magnani, E., Krämer, N., Eschenhagen, R., Rosasco, L., Hennig, P.: Approximate

- Bayesian Neural Operators: Uncertainty Quantification for Parametric PDEs (2022) [arXiv:2208.01565](https://arxiv.org/abs/2208.01565) [cs.LG]
- [83] Hüllermeier, E., Waegeman, W.: Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods. *Machine Learning* **110**(3), 457–506 (2021) <https://doi.org/10.1007/s10994-021-05946-3>
- [84] Gawlikowski, J., Tassi, C.R.N., Ali, M., Lee, J., Humt, M., Feng, J., Kruspe, A., Triebel, R., Jung, P., Roscher, R., Shahzad, M., Yang, W., Bamler, R., Zhu, X.X.: A survey of uncertainty in deep neural networks (2022) [arXiv:2107.03342](https://arxiv.org/abs/2107.03342) [cs.LG]
- [85] Bremner, M.J., Jozsa, R., Shepherd, D.J.: Classical simulation of commuting quantum computations implies collapse of the polynomial hierarchy. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **467**(2126), 459–472 (2010) <https://doi.org/10.1098/rspa.2010.0301>
- [86] Johansson, M.P., Krishnasamy, E., Meyer, N., Piechurski, C.: Quantum computing – a European perspective. PRACE Technical Report (2021)
- [87] Kim, Y., Eddins, A., Anand, S., Wei, K.X., Berg, E., Rosenblatt, S., Nayfeh, H., Wu, Y., Zaletel, M., Temme, K., Kandala, A.: Evidence for the utility of quantum computing before fault tolerance. *Nature* **618**, 500–505 (2023)
- [88] Coyle, B., Mills, D., Danos, V., Kashefi, E.: The born supremacy: quantum advantage and training of an ising born machine. *npj Quantum Information* **6**(1), 60 (2020) <https://doi.org/10.1038/s41534-020-00288-9>
- [89] Liu, Y., Arunachalam, S., Temme, K.: A rigorous and robust quantum speed-up in supervised machine learning. *Nature Physics* **17**(9), 1013–1017 (2021)
- [90] Quek, Y., França, D.S., Khatri, S., Meyer, J.J., Eisert, J.: Exponentially tighter bounds on limitations of quantum error mitigation (arXiv:2210.11505) (2022). [arXiv:2210.11505](https://arxiv.org/abs/2210.11505) [math-ph, physics:quant-ph]
- [91] Stilek França, D., García-Patrón, R.: Limitations of optimization algorithms on noisy quantum devices. *Nature Physics* **17**(11), 1221–1227 (2021) <https://doi.org/10.1038/s41567-021-01356-3>
- [92] ESA EO4Society Website. <https://eo4society.esa.int/projects/qa4eo-study/>

6

Conclusions and Future Works

According to computational complexity conjectures, quantum computers may eventually achieve certain tasks faster than today's conventional classical computers. The qubits required by quantum computers can be implemented in various ways, and a limited number of error-prone qubits are currently available. Quantum computers being available on the market are designed for a specific type and form of computational problems. They are unlikely to fully replace classical computers but will be used along with them. Hybrid classical-quantum approaches are widely studied for tackling real-world problems in industry and academia and for profiting from quantum computers integrated in supercomputers.

Toward quantum computing for Earth observation tasks, we tackle several research questions: 1. which satellite datasets we have to use to gain the advantages and imperfections of deploying QML models on a (future) quantum computer over conventional machine learning techniques, 2. how to embed data points in a small quantum computer, and 3. how to profit from both a supercomputer and quantum computer.

To solve these questions, we identified and proposed satellite datasets to investigate the power of QML approaches when utilizing small-scale faulty quantum computers, namely NISQ computers and QA devices. We also obtained insights into how to profit from an optimal sharing between a supercomputer and quantum computer. Best results are often achieved by exploiting a combination of a classical supercomputer and a quantum computer, known as a hybrid classical-quantum approach.

Chapter 4 of this dissertation assess QML approaches with the help of a conventional classical computer for processing classical input data points. Chapter 5 of this dissertation toward future NISQ computers and FTQCs provides the assessment of quantum computing for EO and practically significant computational problems. We also propose a two-level encoding strategy for embedding class data points in the limited number of input qubits. Our findings demonstrate that QML approaches perform similar to conventional classical machine learning techniques. More importantly, we gain insights into how to program quantum computers for machine learning tasks, and which computational EO problems can be tackled on quantum computers.

Toward future works to gain some computational quantum advantage as early as possible, it is crucial to

1. identify practical computational problems that can be directly implemented on quantum computers, known as quantum-native problems,
2. define application-specific benchmarks to evaluate diverse quantum hardware and algorithms, and
3. build a quantum hardware-agnostic software stack.

Current quantum hardware is prone to errors, so quantum-native problems must take into account the noise in the quantum hardware, especially since quantum error mitigation techniques are not yet perfect. It is even unclear whether quantum error mitigation techniques for

NISQ computers will continue functioning effectively as the quantum system size increases. Progressing from **NISQ** computers to **FTQCs**, the size of the quantum processors, i.e., the number of qubits, is not the only deciding factor. Instead, key issues include the computer's stability, reducing the number of errors using an error mitigation scheme, and the success of the efforts to use logical qubits. However, quantum algorithms typically require subroutines, which are hard to implement on existing **NISQ** computers due to qubit noise. We expect a gradual shift from **NISQ** computers to **FTQCs** once they become available and conclude that we should aim for future **FTQCs** and high-quality **NISQ** computers for practical problems integrated in **HPC** systems when considering full quantum error correction and mitigation schemes.

Bibliography

- [1] Scott Aaronson. “How Much Structure Is Needed for Huge Quantum Speedups?” In: *arXiv* (2022), pp. 1–16. DOI: [10.48550/arXiv.2209.06930](https://doi.org/10.48550/arXiv.2209.06930). URL: <https://doi.org/10.48550/arXiv.2209.06930>.
- [2] Amira Abbas, David Sutter, Christa Zoufal, Aurelien Lucchi, Alessio Figalli, and Stefan Woerner. “The power of quantum neural networks”. In: *Nature Computational Science* 1 (2021), pp. 403–409. ISSN: 2662-8457. DOI: [10.1038/s43588-021-00084-1](https://doi.org/10.1038/s43588-021-00084-1). URL: <https://doi.org/10.1038/s43588-021-00084-1>.
- [3] Oludare Isaac Abiodun, Aman Jantan, Abiodun Esther Omolara, Kemi Victoria Dada, Abubakar Malah Umar, Okafor Uchenwa Linus, Humaira Arshad, Abdullahi Aminu Kazaure, Usman Gana, and Muhammad Ubale Kiru. “Comprehensive Review of Artificial Neural Network Applications to Pattern Recognition”. In: *IEEE Access* 7 (2019), pp. 158820–158846. ISSN: 2169-3536. DOI: [10.1109/ACCESS.2019.2945545](https://doi.org/10.1109/ACCESS.2019.2945545). URL: <https://doi.org/10.1109/ACCESS.2019.2945545>.
- [4] Rajeev Acharya et al. “Suppressing quantum errors by scaling a surface code logical qubit”. In: *Nature* 614 (2023), pp. 676–681. ISSN: 1476-4687. DOI: [10.1038/s41586-022-05434-1](https://doi.org/10.1038/s41586-022-05434-1). URL: <https://doi.org/10.1038/s41586-022-05434-1>.
- [5] Tameem Albash and Daniel A. Lidar. “Adiabatic quantum computation”. In: *Reviews of Modern Physics* 90 (2018). ISSN: 1539-0756. DOI: [10.1103/revmodphys.90.015002](https://dx.doi.org/10.1103/RevModPhys.90.015002). URL: <http://dx.doi.org/10.1103/RevModPhys.90.015002>.
- [6] Sanjeev Arora and Boaz Barak, eds. *Complexity theory: A modern approach*. Cambridge: Cambridge University Press., 2009.
- [7] Ryan Babbush, Peter J. Love, and Alán Aspuru-Guzik. “Adiabatic Quantum Simulation of Quantum Chemistry”. In: *Scientific Reports* 4 (2014), p. 6603. ISSN: 2045-2322. DOI: [10.1038/srep06603](https://doi.org/10.1038/srep06603). URL: <https://doi.org/10.1038/srep06603>.
- [8] Stephen Barnett, ed. *Quantum Information*. Oxford: Oxford university Press., 2009.
- [9] Paul Benioff. “The computer as a physical system: A microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines”. In: *Journal of Statistical Physics* 22 (1980), pp. 563–591. ISSN: 1572-9613. DOI: [10.1007/BF01011339](https://doi.org/10.1007/BF01011339). URL: <https://doi.org/10.1007/BF01011339>.
- [10] Kishor Bharti, Alba Cervera-Lierta, Thi Ha Kyaw, Tobias Haug, Sumner Alperin-Lea, Abhinav Anand, Matthias Degroote, Hermanni Heimonen, Jakob S. Kottmann, Tim Menke, Wai-Keong Mok, Sukin Sim, Leong-Chuan Kwek, and Alán Aspuru-Guzik. “Noisy intermediate-scale quantum algorithms”. In: *Rev. Mod. Phys.* 94 (2022), p. 015004. DOI: [10.1103/RevModPhys.94.015004](https://link.aps.org/doi/10.1103/RevModPhys.94.015004). URL: <https://link.aps.org/doi/10.1103/RevModPhys.94.015004>.
- [11] Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. “Quantum machine learning”. In: *Nature* 549 (2017), pp. 195–202. ISSN: 1476-4687. DOI: [10.1038/nature23474](https://doi.org/10.1038/nature23474). URL: <https://doi.org/10.1038/nature23474>.

- [12] Zhengbing Bian, Fabian Chudak, Robert Brian Israel, Brad Lackey, William G. Macready, and Aidan Roy. “Mapping Constrained Optimization Problems to Quantum Annealing with Application to Fault Diagnosis”. In: *Frontiers in ICT* 3 (2016). ISSN: 2297-198X. DOI: [10.3389/fict.2016.00014](https://doi.org/10.3389/fict.2016.00014). URL: <https://www.frontiersin.org/article/10.3389/fict.2016.00014>.
- [13] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738.
- [14] Christian Blum and Andrea Roli. “Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison”. In: *ACM Comput. Surv.* 35.3 (2003), pp. 268–308. ISSN: 0360-0300. DOI: [10.1145/937503.937505](https://doi.org/10.1145/937503.937505). URL: <https://doi.org/10.1145/937503.937505>.
- [15] John Boyd. “Silicon chip delivers quantum speeds [News]”. In: *IEEE Spectrum* 55 (2018), pp. 10–11. ISSN: 1939-9340. DOI: [10.1109/MSPEC.2018.8389173](https://doi.org/10.1109/MSPEC.2018.8389173). URL: <https://doi.org/10.1109/MSPEC.2018.8389173>.
- [16] Michael Broughton, Guillaume Verdon, Trevor McCourt, Antonio J. Martinez, Jae Hyeon Yoo, Sergei V. Isakov, Philip Massey, Murphy Yuezhen Niu, Ramin Halavati, Evan Peters, Martin Leib, Andrea Skolik, Michael Streif, David Von Dollen, Jarrod R. McClean, Sergio Boixo, Dave Bacon, Alan K. Ho, Hartmut Neven, and Masoud Mohseni. “TensorFlow Quantum: A Software Framework for Quantum Machine Learning”. In: *arXiv* (2020), pp. 1–56. DOI: [10.48550/arXiv.2003.02989](https://doi.org/10.48550/arXiv.2003.02989). URL: <https://doi.org/10.48550/arXiv.2003.02989>.
- [17] Nicolas Brunel, Vincent Hakim, and Magnus JE Richardson. “Single neuron dynamics and computation”. In: *Current Opinion in Neurobiology* 25 (2014). Theoretical and computational neuroscience, pp. 149–155. ISSN: 0959-4388. DOI: <https://doi.org/10.1016/j.conb.2014.01.005>. URL: <https://www.sciencedirect.com/science/article/pii/S0959438814000130>.
- [18] Jun Cai, William G. Macready, and Aidan Roy. “A practical heuristic for finding graph minors”. In: *arXiv* (2014), pp. 1–16. DOI: [10.48550/arXiv.1406.2741](https://doi.org/10.48550/arXiv.1406.2741). URL: <https://doi.org/10.48550/arXiv.1406.2741>.
- [19] Gustau Camps-Valls, Jochem Verrelst, Jordi Munoz-Mari, Valero Laparra, Fernando Mateo-Jimenez, and Jose Gomez-Dans. “A Survey on Gaussian Processes for Earth-Observation Data Analysis: A Comprehensive Investigation”. In: *IEEE Geoscience and Remote Sensing Magazine* 4 (2016), pp. 58–78. ISSN: 2168-6831. DOI: [10.1109/MGRS.2015.2510084](https://doi.org/10.1109/MGRS.2015.2510084). URL: <https://doi.org/10.1109/MGRS.2015.2510084>.
- [20] V. Canivell, P. Forn-Díaz, A. Garcia-Saez, and R. Sagastizabal. “Startup Qilimanjaro—towards a European full-stack coherent quantum annealer platform”. In: *EPJ Quantum Technology* 8 (2021), p. 6. ISSN: 2196-0763. DOI: [10.1140/epjqt/s40507-021-00094-y](https://doi.org/10.1140/epjqt/s40507-021-00094-y). URL: <https://doi.org/10.1140/epjqt/s40507-021-00094-y>.
- [21] M. Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C. Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R. McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, and Patrick J. Coles. “Variational quantum algorithms”. In: *Nature Reviews Physics* 3 (2021), pp. 625–644. ISSN: 2522-5820. DOI: [10.1038/s42254-021-00348-9](https://doi.org/10.1038/s42254-021-00348-9). URL: <https://doi.org/10.1038/s42254-021-00348-9>.
- [22] Rui Chao and Ben W. Reichardt. “Fault-tolerant quantum computation with few qubits”. In: *npj Quantum Information* 4 (2018), p. 42. ISSN: 2056-6387. DOI: [10.1038/s41534-018-0085-z](https://doi.org/10.1038/s41534-018-0085-z). URL: <https://doi.org/10.1038/s41534-018-0085-z>.

- [23] Gong Cheng, Xingxing Xie, Junwei Han, Lei Guo, and Gui-Song Xia. “Remote Sensing Image Scene Classification Meets Deep Learning: Challenges, Methods, Benchmarks, and Opportunities”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 13 (2020), pp. 3735–3756. ISSN: 2151-1535. DOI: [10.1109/JSTARS.2020.3005403](https://doi.org/10.1109/JSTARS.2020.3005403). URL: <https://doi.org/10.1109/JSTARS.2020.3005403>.
- [24] M. Datcu, K. Seidel, and M. Walessa. “Spatial information retrieval from remote-sensing images. I. Information theoretical perspective”. In: *IEEE Transactions on Geoscience and Remote Sensing* 36 (1998), pp. 1431–1445. ISSN: 1558-0644. DOI: [10.1109/36.718847](https://doi.org/10.1109/36.718847). URL: <https://doi.org/10.1109/36.718847>.
- [25] Vasil S. Denchev, Nan Ding, S. V. N. Vishwanathan, and Hartmut Neven. “Robust Classification with Adiabatic Quantum Optimization”. In: *arXiv* (2012), pp. 1–17. DOI: [10.48550/arXiv.1205.1148](https://doi.org/10.48550/arXiv.1205.1148). URL: <https://doi.org/10.48550/arXiv.1205.1148>.
- [26] Pedro Dias, Joao Catalao, and Fernando O. Marques. “Sentinel-1 InSAR data applied to surface deformation in Macaronesia (Canaries and Cape Verde)”. In: *Procedia Computer Science* 138 (2018), pp. 382–387. ISSN: 1877-0509. DOI: [10.1016/j.procs.2018.10.054](https://doi.org/10.1016/j.procs.2018.10.054). URL: <https://doi.org/10.1016/j.procs.2018.10.054>.
- [27] Vedran Dunjko and Hans J Briegel. “Machine learning & artificial intelligence in the quantum domain: a review of recent progress”. In: *Reports on Progress in Physics* 81.7 (2018), p. 074001. DOI: [10.1088/1361-6633/aab406](https://doi.org/10.1088/1361-6633/aab406). URL: <https://doi.org/10.1088/1361-6633/aab406>.
- [28] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. “A Quantum Approximate Optimization Algorithm”. In: (2014). arXiv: [1411.4028](https://arxiv.org/abs/1411.4028) [quant-ph].
- [29] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, Joshua Lapan, Andrew Lundgren, and Daniel Preda. “A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem”. In: *Science* 292 (2001), pp. 472–475. ISSN: 0036-8075. DOI: [10.1126/science.1057726](https://doi.org/10.1126/science.1057726). URL: <https://doi.org/10.1126/science.1057726>.
- [30] Edward Farhi, Jeffrey Goldstone, Sam Gutmann, and Michael Sipser. “Quantum Computation by Adiabatic Evolution”. In: *arXiv* (2000), pp. 1–24. DOI: [10.48550/arXiv.quant-ph/0001106](https://doi.org/10.48550/arXiv.quant-ph/0001106). URL: <https://doi.org/10.48550/arXiv.quant-ph/0001106>.
- [31] Edward Farhi and Aram W Harrow. “Quantum Supremacy through the Quantum Approximate Optimization Algorithm”. In: *arXiv* (2019), pp. 1–23. DOI: [10.48550/arXiv.1602.07674](https://doi.org/10.48550/arXiv.1602.07674). URL: <https://doi.org/10.48550/arXiv.1602.07674>.
- [32] Richard P. Feynman. “Simulating physics with computers”. In: *International Journal of Theoretical Physics* 21 (1982), pp. 467–488. ISSN: 1572-9575. DOI: [10.1007/BF02650179](https://doi.org/10.1007/BF02650179). URL: <https://doi.org/10.1007/BF02650179>.
- [33] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. “Surface codes: Towards practical large-scale quantum computation”. In: *Phys. Rev. A* 86 (2012), p. 032324. DOI: [10.1103/PhysRevA.86.032324](https://doi.org/10.1103/PhysRevA.86.032324). URL: <https://doi.org/10.1103/PhysRevA.86.032324>.
- [34] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. <http://www.deeplearningbook.org>. MIT Press, 2016.

- [35] Daniel Gottesman. “The Heisenberg Representation of Quantum Computers”. In: *arXiv* (1998), pp. 1–16. DOI: [10.48550/arXiv.quant-ph/9807006](https://doi.org/10.48550/arXiv.quant-ph/9807006). URL: <https://doi.org/10.48550/arXiv.quant-ph/9807006>.
- [36] Erica K. Grant and Travis S. Humble. “Adiabatic Quantum Computing and Quantum Annealing”. In: *Oxford University Press* (2020), pp. 1–24. DOI: [10.1093/acrefore/9780190871994.013.32](https://doi.org/10.1093/acrefore/9780190871994.013.32). URL: <https://doi.org/10.1093/acrefore/9780190871994.013.32>.
- [37] Christian Gross and Immanuel Bloch. “Quantum simulations with ultracold atoms in optical lattices”. In: *Science* 357 (2017), pp. 995–1001. DOI: [10.1126/science.aal3837](https://doi.org/10.1126/science.aal3837). URL: <https://doi.org/10.1126/science.aal3837>.
- [38] Irena Hajnsek and Yves-Louis Desnos, eds. *Polarimetric Synthetic Aperture Radar, Principles and Application*. Vol. 25. Remote Sensing and Digital Image Processing. Springer, 2021. URL: <https://elib.dlr.de/145020/>.
- [39] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. “Quantum Algorithm for Linear Systems of Equations”. In: *Phys. Rev. Lett.* 103 (2009), p. 150502. DOI: [10.1103/PhysRevLett.103.150502](https://doi.org/10.1103/PhysRevLett.103.150502). URL: <https://doi.org/10.1103/PhysRevLett.103.150502>.
- [40] Aram W. Harrow and Ashley Montanaro. “Quantum computational supremacy”. In: *Nature* 549 (2017), pp. 203–209. ISSN: 1476-4687. DOI: [10.1038/nature23458](https://doi.org/10.1038/nature23458). URL: <https://doi.org/10.1038/nature23458>.
- [41] Philipp Hauke, Helmut G Katzgraber, Wolfgang Lechner, Hidetoshi Nishimori, and William D Oliver. “Perspectives of quantum annealing: methods and implementations”. In: *Reports on Progress in Physics* 83.5 (2020), p. 054401. DOI: [10.1088/1361-6633/ab85b8](https://doi.org/10.1088/1361-6633/ab85b8). URL: <https://doi.org/10.1088/1361-6633/ab85b8>.
- [42] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition”. In: *arXiv* (2015), pp. 1–16. DOI: [10.48550/arXiv.1512.03385](https://doi.org/10.48550/arXiv.1512.03385). URL: <https://doi.org/10.48550/arXiv.1512.03385>.
- [43] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. “EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 12 (2019), pp. 2217–2226. ISSN: 2151-1535. DOI: [10.1109/JSTARS.2019.2918242](https://doi.org/10.1109/JSTARS.2019.2918242). URL: <https://doi.org/10.1109/JSTARS.2019.2918242>.
- [44] J.J. Hopfield and D.W. Tank. “Computing with neural circuits: a model”. In: *Science* 233 (1986), pp. 625–633. ISSN: 0036-8075. DOI: [10.1126/science.3755256](https://doi.org/10.1126/science.3755256). URL: <https://doi.org/10.1126/science.3755256>.
- [45] Iordanis Kerenidis, Jonas Landman, Alessandro Luongo, and Anupam Prakash. “q-means: A quantum algorithm for unsupervised machine learning”. In: *Advances in Neural Information Processing Systems 2019 (NeurIPS 2019)* (2019), pp. 1–26. DOI: [10.48550/arXiv.1812.03584](https://doi.org/10.48550/arXiv.1812.03584). URL: <https://doi.org/10.48550/arXiv.1812.03584>.
- [46] Youngseok Kim, Andrew Eddins, Sajant Anand, Ken Xuan Wei, Ewout van den Berg, Sami Rosenblatt, Hasan Nayfeh, Yantao Wu, Michael Zaletel, Kristan Temme, and Abhinav Kandala. “Evidence for the utility of quantum computing before fault tolerance”. In: *Nature* 618 (2023), pp. 500–505. ISSN: 1476-4687. DOI: [10.1038/s41586-023-06096-3](https://doi.org/10.1038/s41586-023-06096-3). URL: <https://doi.org/10.1038/s41586-023-06096-3>.

- [47] Diederik P. Kingma and Max Welling. “An Introduction to Variational Autoencoders”. In: *Foundations and Trends® in Machine Learning* 12 (2019), pp. 307–392. ISSN: 1935-8245. DOI: [10.1561/22000000056](https://doi.org/10.1561/22000000056). URL: <http://dx.doi.org/10.1561/22000000056>.
- [48] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. “ImageNet classification with deep convolutional neural networks”. In: *Commun. ACM* 60 (2017), pp. 84–90. ISSN: 0001-0782. DOI: [10.1145/3065386](https://doi.org/10.1145/3065386). URL: <https://doi.org/10.1145/3065386>.
- [49] Jonas Landman. “Quantum Algorithms for Unsupervised Machine Learning and Neural Networks”. In: *arXiv* (2021), pp. 1–192. DOI: [10.48550/arXiv.2111.03598](https://doi.org/10.48550/arXiv.2111.03598). URL: <https://doi.org/10.48550/arXiv.2111.03598>.
- [50] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. “Deep learning”. In: *Nature* 521 (2015), pp. 436–444. ISSN: 1476-4687. DOI: [10.1038/nature14539](https://doi.org/10.1038/nature14539). URL: <https://doi.org/10.1038/nature14539>.
- [51] Shutao Li, Weiwei Song, Leyuan Fang, Yushi Chen, Pedram Ghamisi, and Jón Atli Benediktsson. “Deep Learning for Hyperspectral Image Classification: An Overview”. In: *IEEE Transactions on Geoscience and Remote Sensing* 57.9 (2019), pp. 6690–6709. DOI: [10.1109/TGRS.2019.2907932](https://doi.org/10.1109/TGRS.2019.2907932).
- [52] Xu Liu, Licheng Jiao, and Fang Liu. “PolSF: PolSAR image dataset on San Francisco”. In: *arXiv* (2022), pp. 1–16. DOI: [10.48550/arXiv.1912.07259](https://doi.org/10.48550/arXiv.1912.07259). URL: <https://doi.org/10.48550/arXiv.1912.07259>.
- [53] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. “A ConvNet for the 2020s”. In: *arXiv* (2022), pp. 1–15. DOI: [10.48550/arXiv.2201.03545](https://doi.org/10.48550/arXiv.2201.03545). URL: <https://doi.org/10.48550/arXiv.2201.03545>.
- [54] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. “Quantum principal component analysis”. In: *Nature Physics* 10 (2014), pp. 631–633. ISSN: 1745-2481. DOI: [10.1038/nphys3029](https://doi.org/10.1038/nphys3029). URL: <https://doi.org/10.1038/nphys3029>.
- [55] Seth Lloyd, Maria Schuld, Aroosa Ijaz, Josh Izaac, and Nathan Killoran. “Quantum embeddings for machine learning”. In: *arXiv* (2020), pp. 1–16. DOI: [10.48550/arXiv.2001.03622](https://doi.org/10.48550/arXiv.2001.03622). URL: <https://doi.org/10.48550/arXiv.2001.03622>.
- [56] Andrew Lucas. “Ising formulations of many NP problems”. In: *Frontiers in Physics* 2 (2014), p. 5. ISSN: 2296-424X. DOI: [10.3389/fphy.2014.00005](https://doi.org/10.3389/fphy.2014.00005). URL: <https://doi.org/10.3389/fphy.2014.00005>.
- [57] Yuri Manin, ed. *Computable and Uncomputable*. Sovetskoye Radio: Moscow., 1980.
- [58] Johannes Jakob Meyer. “Fisher Information in Noisy Intermediate-Scale Quantum Applications”. In: *Quantum* 5 (2021), p. 539. ISSN: 2521-327X. DOI: [10.22331/q-2021-09-09-539](https://doi.org/10.22331/q-2021-09-09-539). URL: <https://doi.org/10.22331/q-2021-09-09-539>.
- [59] K. Mitarai, M. Negoro, M. Kitagawa, and K. Fujii. “Quantum circuit learning”. In: *Physical Review A* 98 (2018). ISSN: 2469-9934. DOI: [10.1103/PhysRevA.98.032309](https://doi.org/10.1103/PhysRevA.98.032309). URL: <http://dx.doi.org/10.1103/PhysRevA.98.032309>.
- [60] Giorgos Mountrakis, Jungho Im, and Caesar Ogole. “Support vector machines in remote sensing: A review”. In: *ISPRS Journal of Photogrammetry and Remote Sensing* 66 (2011), pp. 247–259. ISSN: 0924-2716. DOI: [10.1016/j.isprsjprs.2010.11.001](https://doi.org/10.1016/j.isprsjprs.2010.11.001). URL: <https://doi.org/10.1016/j.isprsjprs.2010.11.001>.
- [61] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012. ISBN: 0262018020.

- [62] Michael A. Nielsen and Isaac Chuang. “Quantum Computation and Quantum Information”. In: *American Journal of Physics* 70.5 (2002), pp. 558–559. DOI: [10.1119/1.1463744](https://doi.org/10.1119/1.1463744). URL: <https://doi.org/10.1119/1.1463744>.
- [63] Perry C. Oddo and John D. Bolten. “The Value of Near Real-Time Earth Observations for Improved Flood Disaster Response”. In: *Frontiers in Environmental Science* 7 (2019). ISSN: 2296-665X. DOI: [10.3389/fenvs.2019.00127](https://doi.org/10.3389/fenvs.2019.00127). URL: <https://doi.org/10.3389/fenvs.2019.00127>.
- [64] Soronzonbold Otgonbaatar and Mihai Datcu. “A Quantum Annealer for Subset Feature Selection and the Classification of Hyperspectral Images”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 14 (2021), pp. 7057–7065. ISSN: 2151-1535. DOI: [10.1109/JSTARS.2021.3095377](https://doi.org/10.1109/JSTARS.2021.3095377). URL: <https://doi.org/10.1109/JSTARS.2021.3095377>.
- [65] Soronzonbold Otgonbaatar and Mihai Datcu. “Assembly of a Coreset of Earth Observation Images on a Small Quantum Computer”. In: *Electronics* 10.20 (2021). ISSN: 2079-9292. DOI: [10.3390/electronics10202482](https://doi.org/10.3390/electronics10202482). URL: <https://doi.org/10.3390/electronics10202482>.
- [66] Soronzonbold Otgonbaatar and Mihai Datcu. “Classification of Remote Sensing Images With Parameterized Quantum Gates”. In: *IEEE Geoscience and Remote Sensing Letters* (2021), pp. 1–5. ISSN: 1558-0571. DOI: [10.1109/LGRS.2021.3108014](https://doi.org/10.1109/LGRS.2021.3108014). URL: <https://doi.org/10.1109/LGRS.2021.3108014>.
- [67] Soronzonbold Otgonbaatar and Mihai Datcu. “Natural Embedding of the Stokes Parameters of Polarimetric Synthetic Aperture Radar Images in a Gate-Based Quantum Computer”. In: *IEEE Transactions on Geoscience and Remote Sensing* 60 (2022), pp. 1–8. ISSN: 1558-0644. DOI: [10.1109/TGRS.2021.3110056](https://doi.org/10.1109/TGRS.2021.3110056). URL: <https://doi.org/10.1109/TGRS.2021.3110056>.
- [68] Soronzonbold Otgonbaatar and Mihai Datcu. “Quantum annealer for network flow minimization in InSAR images”. In: *EUSAR 2021; 13th European Conference on Synthetic Aperture Radar*. 2021, pp. 1–4. URL: <https://ieeexplore.ieee.org/document/9472581>.
- [69] Soronzonbold Otgonbaatar and Mihai Datcu. “Quantum Annealing Approach: Feature Extraction and Segmentation of Synthetic Aperture Radar Image”. In: *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*. 2020, pp. 3692–3695. DOI: [10.1109/IGARSS39084.2020.9323504](https://doi.org/10.1109/IGARSS39084.2020.9323504).
- [70] Soronzonbold Otgonbaatar, Mihai Datcu, and Begüm Demir. “Causality for Remote Sensing: An Exploratory Study”. In: *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*. 2022, pp. 259–262. DOI: [10.1109/IGARSS46834.2022.9883060](https://doi.org/10.1109/IGARSS46834.2022.9883060). URL: <https://doi.org/10.1109/IGARSS46834.2022.9883060>.
- [71] Soronzonbold Otgonbaatar, Mihai Datcu, and Begüm Demir. “Coreset of Hyperspectral Images on a Small Quantum Computer”. In: *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*. 2022, pp. 4923–4926. DOI: [10.1109/IGARSS46834.2022.9884273](https://doi.org/10.1109/IGARSS46834.2022.9884273). URL: <https://doi.org/10.1109/IGARSS46834.2022.9884273>.
- [72] Soronzonbold Otgonbaatar and Dieter Kranzlmüller. “Exploiting the Quantum Advantage for Satellite Image Processing: Review and Assessment”. In: *IEEE Transactions on Quantum Engineering* 5 (2024), pp. 1–9. ISSN: 2689-1808. DOI: [10.1109/TQE.2023.3338970](https://doi.org/10.1109/TQE.2023.3338970). URL: <https://doi.org/10.1109/TQE.2023.3338970>.

- [73] Soronzonbold Otgonbaatar and Dieter Kranzlmüller. “Quantum-inspired Tensor Network for Earth Science”. In: *IGARSS 2023 - 2023 IEEE International Geoscience and Remote Sensing Symposium*. 2023, pp. 788–791. DOI: [10.1109/IGARSS52108.2023.10282577](https://doi.org/10.1109/IGARSS52108.2023.10282577). URL: <https://doi.org/10.1109/IGARSS52108.2023.10282577>.
- [74] Soronzonbold Otgonbaatar, Olli Nurmi, Mikael Johansson, Jarmo Mäkelä, tadeusz kocman tadeusz, Piotr Gawron, Zbigniew Puchala, Jakub Mielzcarek, Artur Miroszewski, and Corneliu Octavian Dumitru. “Quantum Computing for Climate Change Detection, Climate Modeling, and Climate Digital Twins”. In: *preparation for Nature Reviews Earth and Environment* (2023). DOI: [10.36227/techrxiv.24478663.v1](https://doi.org/10.36227/techrxiv.24478663.v1). URL: <http://dx.doi.org/10.36227/techrxiv.24478663.v1>.
- [75] Soronzonbold Otgonbaatar, Gottfried Schwarz, Mihai Datcu, and Dieter Kranzlmüller. “Quantum Transfer Learning for Real-World, Small, and High-Dimensional Remotely Sensed Datasets”. In: *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 16 (2023), pp. 9223–9230. ISSN: 2151-1535. DOI: [10.1109/JSTARS.2023.3316306](https://doi.org/10.1109/JSTARS.2023.3316306). URL: <https://doi.org/10.1109/JSTARS.2023.3316306>.
- [76] Antonio Pepe and Fabiana Calò. “A Review of Interferometric Synthetic Aperture RADAR (InSAR) Multi-Track Approaches for the Retrieval of Earth’s Surface Displacements”. In: *Applied Sciences* 7 (2017). ISSN: 2076-3417. DOI: [10.3390/app7121264](https://doi.org/10.3390/app7121264). URL: <https://doi.org/10.3390/app7121264>.
- [77] Claudio Persello, Jan Dirk Wegner, Ronny Hänsch, Devis Tuia, Pedram Ghamisi, Mila Koeva, and Gustau Camps-Valls. “Deep Learning and Earth Observation to Support the Sustainable Development Goals”. In: *arXiv* (2022), pp. 1–23. DOI: [10.48550/arXiv.2112.11367](https://doi.org/10.48550/arXiv.2112.11367). URL: <https://doi.org/10.48550/arXiv.2112.11367>.
- [78] John Preskill. “Fault-tolerant quantum computation”. In: *arXiv* (1997), pp. 1–58. DOI: [10.48550/arXiv.quant-ph/9712048](https://doi.org/10.48550/arXiv.quant-ph/9712048). URL: <https://doi.org/10.48550/arXiv.quant-ph/9712048>.
- [79] John Preskill. “Quantum computing 40 years later”. In: *arXiv* (1997), pp. 1–58. DOI: [10.48550/arXiv.2106.10522](https://doi.org/10.48550/arXiv.2106.10522). URL: <https://doi.org/10.48550/arXiv.2106.10522>.
- [80] John Preskill. “Quantum Computing in the NISQ era and beyond”. In: *Quantum* 2 (2018), p. 79. ISSN: 2521-327X. DOI: [10.22331/q-2018-08-06-79](https://doi.org/10.22331/q-2018-08-06-79). URL: <https://doi.org/10.22331/q-2018-08-06-79>.
- [81] Patrick Rebentrost, Masoud Mohseni, and Seth Lloyd. “Quantum Support Vector Machine for Big Data Classification”. In: *Phys. Rev. Lett.* 113 (2014), p. 130503. DOI: [10.1103/PhysRevLett.113.130503](https://doi.org/10.1103/PhysRevLett.113.130503). URL: <https://doi.org/10.1103/PhysRevLett.113.130503>.
- [82] Markus Reichstein, Gustau Camps-Valls, Bjorn Stevens, Martin Jung, Joachim Denzler, Nuno Carvalhais, and Prabhat. “Deep learning and process understanding for data-driven Earth system science”. In: *Nature* 566 (2019), pp. 195–204. ISSN: 1476-4687. DOI: [10.1038/s41586-019-0912-1](https://doi.org/10.1038/s41586-019-0912-1). URL: <https://doi.org/10.1038/s41586-019-0912-1>.
- [83] Markus Reiher, Nathan Wiebe, Krysta M. Svore, Dave Wecker, and Matthias Troyer. “Elucidating reaction mechanisms on quantum computers”. In: *Proceedings of the National Academy of Sciences* 114.29 (2017), pp. 7555–7560. DOI: [10.1073/pnas.1619152114](https://doi.org/10.1073/pnas.1619152114). eprint: <https://www.pnas.org/doi/pdf/10.1073/pnas.1619152114>. URL: <https://www.pnas.org/doi/abs/10.1073/pnas.1619152114>.

- [84] John Richards. “Remote Sensing Digital Image Analysis”. In: *Berlin* (2013). DOI: [10.1007/978-3-642-30062-2](https://doi.org/10.1007/978-3-642-30062-2). URL: <https://doi.org/10.1007/978-3-642-30062-2>.
- [85] V. Saggio, B. E. Asenbeck, A. Hamann, T. Strömberg, P. Schiansky, V. Dunjko, N. Friis, N. C. Harris, M. Hochberg, D. Englund, S. Wölk, H. J. Briegel, and P. Walther. “Experimental quantum speed-up in reinforcement learning agents”. In: *Nature* 591 (2021), pp. 229–233. ISSN: 1476-4687. DOI: [10.1038/s41586-021-03242-7](https://doi.org/10.1038/s41586-021-03242-7). URL: <https://doi.org/10.1038/s41586-021-03242-7>.
- [86] Ruslan Salakhutdinov and Geoffrey Hinton. “Deep Boltzmann Machines”. In: vol. 5. *Proceedings of Machine Learning Research*. Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA: PMLR, 2009, pp. 448–455. URL: <http://proceedings.mlr.press/v5/salakhutdinov09a.html>.
- [87] Bernhard Schölkopf, Alex J. Smola, Robert C. Williamson, and Peter L. Bartlett. “New Support Vector Algorithms”. In: *Neural Comput.* 12.5 (2000), pp. 1207–1245. ISSN: 0899-7667. DOI: [10.1162/089976600300015565](https://doi.org/10.1162/089976600300015565). URL: <https://doi.org/10.1162/089976600300015565>.
- [88] M. Schroder, H. Rehrauer, K. Seidel, and M. Datcu. “Spatial information retrieval from remote-sensing images. II. Gibbs-Markov random fields”. In: *IEEE Transactions on Geoscience and Remote Sensing* 36 (1998), pp. 1446–1455. ISSN: 1558-0644. DOI: [10.1109/36.718848](https://doi.org/10.1109/36.718848). URL: <https://doi.org/10.1109/36.718848>.
- [89] Maria Schuld and Nathan Killoran. “Quantum Machine Learning in Feature Hilbert Spaces”. In: *Phys. Rev. Lett.* 122 (2019), p. 040504. DOI: [10.1103/PhysRevLett.122.040504](https://doi.org/10.1103/PhysRevLett.122.040504). URL: <https://doi.org/10.1103/PhysRevLett.122.040504>.
- [90] Maria Schuld, Ryan Sweke, and Johannes Jakob Meyer. “Effect of data encoding on the expressive power of variational quantum-machine-learning models”. In: *Phys. Rev. A* 103 (2021), p. 032430. DOI: [10.1103/PhysRevA.103.032430](https://doi.org/10.1103/PhysRevA.103.032430). URL: <https://doi.org/10.1103/PhysRevA.103.032430>.
- [91] Yilei Shi, Xiao xiang Zhu, and Richard Bamler. “Nonlocal Compressive Sensing-Based SAR Tomography”. In: *IEEE Transactions on Geoscience and Remote Sensing* 57 (2019), pp. 3015–3024. ISSN: 1558-0644. DOI: [10.1109/TGRS.2018.2879382](https://doi.org/10.1109/TGRS.2018.2879382). URL: <https://doi.org/10.1109/TGRS.2018.2879382>.
- [92] Peter W. Shor. “Fault-tolerant quantum computation”. In: *arXiv* (1997), pp. 1–11. DOI: [10.48550/arXiv.quant-ph/9605011](https://doi.org/10.48550/arXiv.quant-ph/9605011). URL: <https://doi.org/10.48550/arXiv.quant-ph/9605011>.
- [93] Peter W. Shor. “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”. In: *SIAM Review* 41 (1999), pp. 303–332. DOI: [10.1137/S0036144598347011](https://doi.org/10.1137/S0036144598347011). URL: <https://doi.org/10.1137/S0036144598347011>.
- [94] Karen Simonyan and Andrew Zisserman. “Very Deep Convolutional Networks for Large-Scale Image Recognition”. In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: <http://arxiv.org/abs/1409.1556>.
- [95] Andrea Skolik, Jarrod R. McClean, Masoud Mohseni, Patrick van der Smagt, and Martin Leib. “Layerwise learning for quantum neural networks”. In: *Quantum Machine Intelligence* 3 (2021), p. 5. ISSN: 2524-4914. DOI: [10.1007/s42484-020-00036-4](https://doi.org/10.1007/s42484-020-00036-4). URL: <https://doi.org/10.1007/s42484-020-00036-4>.

- [96] T. Stollenwerk, B. O’Gorman, D. Venturelli, S. Mandrà, O. Rodionova, H. Ng, B. Sridhar, E. G. Rieffel, and R. Biswas. “Quantum Annealing Applied to De-Conflicting Optimal Trajectories for Air Traffic Management”. In: *IEEE Transactions on Intelligent Transportation Systems* 21 (2020), pp. 285–297. ISSN: 1558-0016. DOI: [10.1109/TITS.2019.2891235](https://doi.org/10.1109/TITS.2019.2891235). URL: <https://doi.org/10.1109/TITS.2019.2891235>.
- [97] Michael Streif. “Exploration of quantum algorithms for optimization problems on small-scale quantum computers”. PhD thesis. Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 2022. URL: <https://open.fau.de/handle/openfau/18045>.
- [98] Ana Stritih, Peter Bebi, and Adrienne Grêt-Regamey. “Quantifying uncertainties in earth observation-based ecosystem service assessments”. In: *Environmental Modelling and Software* 111 (2019), pp. 300–310. ISSN: 1364-8152. DOI: [10.1016/j.envsoft.2018.09.005](https://doi.org/10.1016/j.envsoft.2018.09.005). URL: <https://doi.org/10.1016/j.envsoft.2018.09.005>.
- [99] Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. “Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning”. In: *arXiv* (2022), pp. 1–12. DOI: [10.48550/arXiv.1602.07261](https://doi.org/10.48550/arXiv.1602.07261). URL: <https://doi.org/10.48550/arXiv.1602.07261>.
- [100] Devis Tuia, Konrad Schindler, Begüm Demir, Gustau Camps-Valls, Xiao Xiang Zhu, Mrinalini Kochupillai, Sašo Džeroski, Jan N. van Rijn, Holger H. Hoos, Fabio Del Frate, Mihai Datcu, Jorge-Arnulfo Quiané-Ruiz, Volker Markl, Bertrand Le Saux, and Rochelle Schneider. “Artificial intelligence to advance Earth observation: a perspective”. In: *arXiv* (2022), pp. 1–16. DOI: [10.48550/arXiv.2305.08413](https://doi.org/10.48550/arXiv.2305.08413). URL: <https://doi.org/10.48550/arXiv.2305.08413>.
- [101] G Wendin. “Quantum information processing with superconducting circuits: a review”. In: *Reports on Progress in Physics* 80.10 (2017), p. 106001. DOI: [10.1088/1361-6633/aa7e1a](https://doi.org/10.1088/1361-6633/aa7e1a). URL: <https://doi.org/10.1088/1361-6633/aa7e1a>.
- [102] D. Willsch, M. Willsch, H. De Raedt, and K. Michielsen. “Support vector machines on the D-Wave quantum annealer”. In: *Computer Physics Communications* 248 (2020), p. 107006. ISSN: 0010-4655. DOI: <https://doi.org/10.1016/j.cpc.2019.107006>. URL: <https://www.sciencedirect.com/science/article/pii/S001046551930342X>.
- [103] Yulin et. al. Wu. “Strong Quantum Computational Advantage Using a Superconducting Quantum Processor”. In: *Phys. Rev. Lett.* 127 (2021), p. 180501. DOI: [10.1103/PhysRevLett.127.180501](https://doi.org/10.1103/PhysRevLett.127.180501). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.127.180501>.
- [104] Hiroshi Yano, Yudai Suzuki, Kohei Itoh, Rudy Raymond, and Naoki Yamamoto. “Efficient Discrete Feature Encoding for Variational Quantum Classifier”. In: *IEEE Transactions on Quantum Engineering* 2 (2021), pp. 1–14. ISSN: 2689-1808. DOI: [10.1109/tqe.2021.3103050](https://doi.org/10.1109/tqe.2021.3103050). URL: <http://dx.doi.org/10.1109/TQE.2021.3103050>.
- [105] Matthew D Zeiler and Rob Fergus. “Visualizing and Understanding Convolutional Networks”. In: *arXiv* (2013), pp. 1–11. DOI: [10.48550/arXiv.1311.2901](https://doi.org/10.48550/arXiv.1311.2901). URL: <https://doi.org/10.48550/arXiv.1311.2901>.

- [106] Han-Sen Zhong, Yu-Hao Deng, Jian Qin, Hui Wang, Ming-Cheng Chen, Li-Chao Peng, Yi-Han Luo, Dian Wu, Si-Qiu Gong, Hao Su, Yi Hu, Peng Hu, Xiao-Yan Yang, Wei-Jun Zhang, Hao Li, Yuxuan Li, Xiao Jiang, Lin Gan, Guangwen Yang, Lixing You, Zhen Wang, Li Li, Nai-Le Liu, Jelmer J. Renema, Chao-Yang Lu, and Jian-Wei Pan. “Phase-Programmable Gaussian Boson Sampling Using Stimulated Squeezed Light”. In: *Phys. Rev. Lett.* 127 (2021), p. 180502. DOI: [10.1103/PhysRevLett.127.180502](https://doi.org/10.1103/PhysRevLett.127.180502). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.127.180502>.
- [107] Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin. “Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices”. In: *Physical Review X* 10 (2020). ISSN: 2160-3308. DOI: [10.1103/physrevx.10.021067](https://doi.org/10.1103/physrevx.10.021067). URL: <http://dx.doi.org/10.1103/PhysRevX.10.021067>.
- [108] Xiao Xiang Zhu, Sina Montazeri, Mohsin Ali, Yuansheng Hua, Yuanyuan Wang, Lichao Mou, Yilei Shi, Feng Xu, and Richard Bamler. “Deep Learning Meets SAR: Concepts, models, pitfalls, and perspectives”. In: *IEEE Geoscience and Remote Sensing Magazine* 9 (2021), pp. 143–172. DOI: [10.1109/MGRS.2020.3046356](https://doi.org/10.1109/MGRS.2020.3046356). URL: <https://doi.org/10.1109/MGRS.2020.3046356>.
- [109] Christa Zoufal. “Generative Quantum Machine Learning”. In: *arXiv* (2021), pp. 1–127. DOI: [10.48550/arXiv.2111.12738](https://doi.org/10.48550/arXiv.2111.12738). URL: <https://doi.org/10.48550/arXiv.2111.12738>.

Webography

- [W1] *German Hyperspectral Satellite*. last accessed in 2024. URL: <https://www.enmap.org/mission/>.
- [W2] *Airborne visible/infrared imaging spectrometer (AVIRIS)*. last accessed in 2024. URL: <https://aviris.jpl.nasa.gov/>.
- [W3] *Quantum algorithm zoo*. last accessed in 2024. URL: <https://quantumalgorithmzoo.org>.
- [W4] *Quantum Digital Twins*. last accessed in 2024. URL: <https://github.com/sozoluffy/Quantum-Digital-Twins/tree/main>.
- [W5] *Convolutional Neural Networks*. last accessed in 2024. URL: <https://cs231n.github.io/convolutional-networks/>.
- [W6] *IBM qiskit*. last accessed in 2024. URL: https://qiskit.org/documentation/tutorials/circuits/3_summary_of_quantum_operations.html.
- [W7] *What is remote sensing?* last accessed in 2024. URL: <https://earthdata.nasa.gov/learn/backgrounders/remote-sensing>.
- [W8] *IBM quantum experience*. last accessed in 2024. URL: <https://quantum-computing.ibm.com/>.
- [W9] *A digital quantum annealer: Qilimanjaro Tech*. last accessed in 2024. URL: <https://www.qilimanjaro.tech/>.
- [W10] *A D-Wave quantum annealer*. last accessed in 2024. URL: <https://cloud.dwavesys.com/leap>.

Acronyms

- AI** Artificial Intelligence 1–5, 9, 13, 18, 19, 22, 25, 26, 72
- ANN** Artificial Neural Networks 9–11, 18, 19
- BI** Bayesian Inference 9
- BM** Boltzmann Machines 9
- BQP** Bounded-Error Quantum Polynomial 2
- CNN** Convolutional Neural Network 3, 11–13, 18, 19
- ConvNet or CNN** Convolutional Network 11
- CRedit** Contributor Roles Taxonomy 5
- CSVM** Classical Support Vector Machine 72
- DL** Deep Learning 1, 10, 31, 87
- DNN** Deep Neural Networks 9, 20
- ECOC** Error Correcting Output Code 62
- EO** Earth Observation 1, 3–5, 9, 10, 25, 26, 31, 56, 57, 138
- FCNL** Fully-Connected Neuron Layer 12
- FTQC** Fault-Tolerant Quantum Computer 4, 138, 139
- GPU** Graphics Processing Unit 87
- HHL** Harrow-Hassidim-Lloyd 13, 18
- HPC** High-Performance Computing 86, 92, 102, 139
- HSI** Hyperspectral Image 2, 23
- InSAR** Interferometric Synthetic Aperture Radar 2, 23, 57
- ML** Machine Learning 1, 4, 18, 21, 47
- NISQ** Noisy Intermediate-Scale Quantum 3, 4, 9, 18, 26, 28, 31, 72, 86, 138, 139
- NP** Non-Deterministic Polynomial 2, 4, 29, 62
- NP-complete** Non-Deterministic Polynomial Complete 2

- P** Polynomial 2
- PDE** Partial Differential Equation 87
- PINNs** Physics-Informed Neural Networks 87
- PolSAR** Polarimetric Synthetic Aperture Radar 2, 23, 24, 47, 72
- PQC** Parameterized Quantum Circuit 4, 18–20, 24, 27
- QA** Quantum Annealing 3–5, 9, 25, 26, 31, 56, 57, 72, 86, 138
- QAI** Quantum Artificial Intelligence 9, 13, 18
- Qboost** Quantum boost 62
- QC** Quantum Computing 1, 9, 13, 18, 26, 92, 102
- QML** Quantum Machine Learning 1, 3, 4, 9, 18, 22, 24, 26, 31, 32, 38, 47, 87, 92, 102, 138
- QNN** Quantum Neural Network 3, 13, 18
- QP** Quadratic Programming 29, 30
- QPCA** Quantum Principal Component Analysis 18
- QSVM** Quantum Support Vector Machine 3, 4, 18, 21, 72
- QUBO** Quadratic Unconstrained Binary Optimization 3, 21, 25, 28–30, 56, 57, 62
- ReLU** Rectified Linear Unit 11–13
- SAR** Synthetic Aperture Radar 1, 22, 23
- SVM** Support Vector Machines 1, 3, 9, 25, 29, 30, 62, 72
- VQAs** Variational Quantum Algorithms 9, 18–20, 26, 27

List of Figures

1.1	A computational complexity map	3
1.2	The causal structure of this dissertation	5
2.1	An artificial neural network	10
2.2	Activation functions for neural networks	11
2.3	A convolutional neural network architecture	12
2.4	The Bloch sphere representation	14
2.5	The decomposition of a Toffoli gate via a universal gate set	17
2.6	A circuit representation of variational quantum algorithms	19
2.7	An L-depth parameterized quantum circuit	20
2.8	A variational quantum model having three input qubits	20
2.9	Example Eurosat images	22
2.10	An Example UC Merced Land Use dataset	22
2.11	Hyperspectral images	23
2.12	An InSAR image of the Cape Verde volcano	23
2.13	Polarimetric synthetic aperture radar images	24
3.1	A hybrid classical-quantum approach	27
3.2	The Pegasus topology of the D-Wave quantum annealer	28

List of Tables

2.1	Single-qubit quantum gates	15
2.2	Multi-qubit quantum gates	16