



# Quantum Computing Applications in Future Colliders

Heather M. Gray<sup>1,2\*</sup> and Koji Terashi<sup>3</sup>

<sup>1</sup>Department of Physics, University of California, Berkeley, Berkeley, CA, United States, <sup>2</sup>Physics Division, Lawrence Berkeley National Laboratory, Berkeley, CA, United States, <sup>3</sup>International Center for Elementary Particle Physics (ICEPP), The University of Tokyo, Tokyo, Japan

High-energy physics is facing a daunting computing challenge with the large amount of data expected from the HL-LHC and other future colliders. In addition, the landscape of computation has been expanding dramatically with technologies beyond the standard x86 CPU architecture becoming increasingly available. Both of these factors necessitate an extensive and broad-ranging research and development campaign. As quantum computation has been evolving rapidly over the past few years, it is important to evaluate how quantum computation could be one potential avenue for development for future collider experiments. A wide variety of applications have been considered by different authors. We review here selected applications of quantum computing to high-energy physics, including topics in simulation, reconstruction, and the use of machine learning, and their challenges. In addition, recent advances in quantum computing technology to enhance such applications are briefly highlighted. Finally, we will discuss how such applications might transform the workflows of future collider experiments and highlight other potential applications.

**Keywords:** quantum computing, future collider experiments, quantum machine learning, quantum annealers, digital quantum computer, pattern recognition

## OPEN ACCESS

### Edited by:

Petra Merkel,  
Fermi National Accelerator Laboratory  
(DOE), United States

### Reviewed by:

Jean-Roch Vlimant,  
California Institute of Technology,  
United States  
Gabriel Nathan Perdue,  
Fermi National Accelerator Laboratory  
(DOE), United States

### \*Correspondence:

Heather M. Gray  
heather.gray@berkeley.edu

### Specialty section:

This article was submitted to  
Radiation Detectors and Imaging,  
a section of the journal  
Frontiers in Physics

**Received:** 28 January 2022

**Accepted:** 03 May 2022

**Published:** 27 May 2022

### Citation:

Gray HM and Terashi K (2022)  
Quantum Computing Applications in  
Future Colliders.  
Front. Phys. 10:864823.  
doi: 10.3389/fphy.2022.864823

## 1 INTRODUCTION

The fields of particle physics and computing have long been intertwined. The success of particle physics depends on the use of cutting-edge computing technology and in certain cases, the requirements of particle physics experiments have stimulated the development of new technologies in computing. Perhaps the most notable example is the development of the world wide web by physicists at the European Council for Nuclear Research (CERN), see [1], and other examples include the introduction of distributed grid computing in [2].

The state-of-the-art collider in particle physics is the Large Hadron Collider (LHC) [3], which is located at CERN just outside Geneva in Switzerland. The first proton-proton collisions in the LHC were recorded in 2010 and a key achievement was the discovery of the Higgs boson in 2012 by the two general purpose experiments, ATLAS and CMS. The third data-taking run of the LHC (Run 3) is planned to start in 2022 and to continue for four years. Run 3 is scheduled to be followed by a three year long shutdown during which the accelerator and experiments will undergo significant upgrades. The high-luminosity LHC (HL-LHC), as discussed in [4], is expected to start delivering proton-proton collisions in 2029.

The HL-LHC will deliver collisions at an instantaneous luminosity a factor of five to seven higher than the original LHC design luminosity. These collisions will increase the number of additional interactions, or pile up, by up to a factor of five to reach approximately 140–200 average pile up

interactions per bunch crossing. The experiments will include new detectors with more readout channels, which will increase the size of the recorded events by a factor of four to five. In addition, upgrades to the trigger systems will increase the event rate by up to an order of magnitude. These extensive upgrades herald the start of a decades long program in precision and discovery physics. At the same time, they place strong demands on computing.

Current projections for future computing budgets for the HL-LHC follow the so-called flat-budget scenario, in which only small increases to the budgets are foreseen to account for expectations from inflation. Extrapolations of the current computing model to the HL-LHC show a large deficit compared to the requirements [5–8]. In addition, computing for particle physics experiments has relied on Moore's Law. Moore's Law is the observation that the number of transistors on integrated circuits doubles every two years. However, over the past decade processor speeds have become limited by power density such that speed increases are due to increases in the number of cores rather than the speed of individual cores. In addition, the hardware landscape for computing processors has become increasingly heterogeneous. A range of development efforts are ongoing to explore how HEP software can be adapted to efficiently exploit heterogeneous computing architectures.

Although we have not yet entered the HL-LHC era, given the timescales required to build accelerators and detectors for particle physics experiments, the field is undergoing a series of international and national review processes to determine the future collider facilities to follow the HL-LHC. Although consensus has not yet been reached, many of the future colliders under consideration would make even more extensive demands on computing. One such collider, the Future Circular Collider in [9] would collide protons at a center-of-mass energy of 100 TeV with up to a thousand pile up interactions per bunch crossing.

The ideas of exploiting quantum mechanics to build a computer first began to be explored more than four decades ago. Initial ideas were focused on how a quantum computer could be used to simulate quantum mechanical systems. A decade later further interest was stimulated when quantum algorithms, which could be used to solve classically intractable problems, were introduced. One of the earliest of these, and one of the most famous, is Shor's algorithm for the factorization of prime numbers. At approximately the same time, the first quantum computers were built based on existing techniques from nuclear magnetic resonance. It is sometimes said that we are currently in the Noisy Intermediate-Scale Quantum (NISQ) era as introduced in [10]. Quantum computers in the NISQ era have orders of tens to hundreds physical qubits, and have been shown to surpass classical computers but only for specifically constructed problems. They also experience significant noise associated with the hardware and the electronics for qubit control.

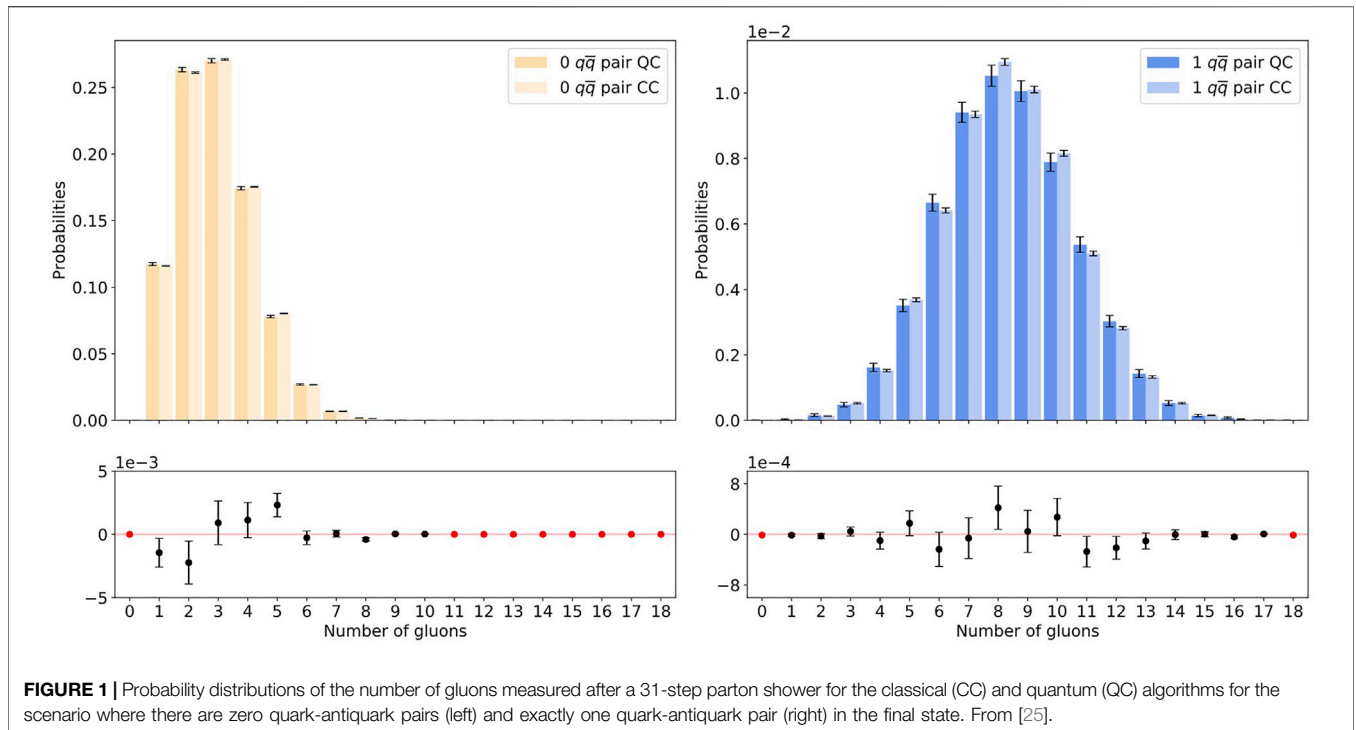
There are two types of quantum computer: quantum annealers (QA) and circuit-based quantum computers. Quantum annealers are specifically designed to solve a single class of problem: minimization problems, and, in particular, to minimization

problems that can be expressed as quadratic unconstrained binary optimization (QUBO) problems. Circuit-based quantum computers, on the other hand, can be programmed to execute more general quantum circuits, and are more similar in concept to the classical computers. Quantum computers use a number of different technologies for the qubits including superconducting transmon qubits, ion traps, photons and topological qubits and their current status is discussed in [11, 12]. A typical state-of-the-art quantum annealer is the one produced by D-Wave with up to 5,000 qubits, while typical state-of-the-art circuit-based quantum computers are those from IBM and Google which have  $\mathcal{O}(60-120)$  superconducting transmon qubits. Thanks to the extensive investment from governments and companies into the development of quantum computers, their capabilities have increased rapidly over the past years. See [13] for a recent review of the field of quantum computation.

Given the computational challenges faced by high-energy physics [14] now and in the future, and given the rapid development and exciting potential of quantum computers, it is natural to ask whether quantum computers can play a role in the future computing at HEP. Fault-tolerant quantum computers with sufficient number of qubits and gates are still decades away and very likely beyond the HL-LHC era, yet, future high-energy physics experiments will place even more extensive demands on computing. This article discusses selected studies exploring the application of quantum computers to HEP. In **Section 2** we discuss how quantum computers might be used to improve the quality of simulation for high-energy physics events. In **Section 3** we discuss one of the key computational challenges in HEP, the reconstruction of charged particle trajectories, and explore how quantum computing could be used for such problems. Finally, in **Section 4** we discuss the exciting field of machine learning with quantum computers and focus on applications of such techniques to physics analysis. **Section 5** concludes the article with a short summary and future outlook.

## 2 APPLICATIONS OF QUANTUM COMPUTING TO SIMULATION

One of the most promising applications for quantum computing is to simulate inherently quantum-mechanical systems, such as systems described by Quantum Field Theory (QFT) in particle physics. Quantum algorithms may perform particle scatterings in QFTs in polynomial resources using a universal quantum computer, as proposed by Jordan, Lee and Preskill in [15] and Preskill in [16]. Since then, a number of pioneering studies have been done in the contexts of simulation of particle systems, e.g., neutrino or neutral Kaon oscillations [17–19], heavy-ion collisions [20], parton distributions inside proton [21] as well as low-energy effective field theory [22] and quantum electrodynamics [23]. However, as represented by the Jordan-Lee-Preskill algorithm, a full QFT simulation generally requires prohibitively large resources and therefore cannot be implemented on near-term quantum computers. An



**FIGURE 1** | Probability distributions of the number of gluons measured after a 31-step parton shower for the classical (CC) and quantum (QC) algorithms for the scenario where there are zero quark-antiquark pairs (left) and exactly one quark-antiquark pair (right) in the final state. From [25].

alternative approach for quantum computing applications of QFT is to break down particle scattering processes into pieces and exploit quantum computations in the place where conventional classical calculations are intractable.

In Monte Carlo simulations of high-energy hadron collisions, physics processes are treated by factorizing hard scatterings which occur at short distances and parton evolutions which occur at long distances inside hadrons. This factorization property allows the short- and long-range subprocesses to be calculated independently. The hard scattering between partons creates a large momentum transfer and produces a cascade of outgoing partons called the parton shower (PS). The PS simulation is a prototypical Markov Chain Monte Carlo (MCMC) simulation and describes the evolution of the system from the hard interaction to the hadronization scale. This technique for PS simulation has been successfully validated by comparing the MC simulation with experimental data, but quantum interference and correlation effects are neglected. Even though these effects are small compared to the current experimental accuracy, such limitations may become a bottleneck when the measurement will reach an unprecedented precision in the HL-LHC era.

Ref. [24] develops a quantum circuit that describes quantum properties of parton showers, in particular, the quantum interference that arises from different intermediate particles using a simplified QFT. They start with a system of  $n$  fermions that can have either one of two flavors,  $f_1$  and  $f_2$ . These fermions can radiate a scalar particle  $\phi$ , which itself can split into a  $f\bar{f}$  pair of the same flavor or different flavors ( $\bar{f}$  is an antifermion). In this setup, [24] demonstrates that quantum interference effects are successfully simulated using a quantum

simulator and hardware, when the  $\phi$  coupling to a  $f_1\bar{f}_2$  or  $\bar{f}_1f_2$  pair is turned on, due to unobserved intermediate states with mixed fermion flavors. This behavior is not properly captured by a classical MCMC simulation. It is also shown that the quantum circuit has capability to sample parton showers from a full probability distribution in polynomial time, which is otherwise difficult with the classical MCMC approach.

Another approach based on a quantum random walk is used in [25] to simulate parton showers. During a random walk, the movement of a particle termed *walker* is controlled by a coin flip operation that determines the direction the walker will move and a shift operation which moves the walker to the next position. The quantum analogue of the random walk performs these operations in a superposition of the basis states for the coin and shift operations, therefore allowing all possible shower histories to be generated when applied to the parton shower. In [25], the emission probabilities are controlled by the coin flip operations, and updating the shower content with a given emission corresponds to the shift operation of the walker in the position space. With this novel approach, [25] manages to simulate a collinear 31-step parton shower, implemented as a two-dimensional quantum walk with gluons and a quark-antiquark pair, as shown in **Figure 1**. This figure shows the probability distributions of the number of gluons after a 31-step parton shower for the classical and quantum algorithms, for the scenario where there are zero (left) and exactly one (right) quark-antiquark pair in the final state. A very good agreement is seen between the two algorithms. The quantum random walk approach demonstrates a significant improvement in the shower depth with fewer number of qubits from the algorithm in [24].

The calculation of the hard scattering also requires significant computational resources in the conventional techniques based on squaring the scattering amplitudes. [26] performs the calculation of hard interactions *via* helicity amplitudes by exploiting the equivalence between qubits and helicity spinors. This relies on the mapping between angles used to parameterize the helicity spinors and the qubit degrees of freedom in the Bloch sphere representation. The operators acting on the spinors are encoded as quantum circuits of unitary operators. With these helicity-qubit encodings, [26] demonstrates the construction of two quantum algorithms. The first is the helicity amplitude calculation in the  $q \rightarrow qg$  process and the second are the helicity calculations for the  $s$ - and  $t$ -channel amplitudes of a  $q\bar{q} \rightarrow q\bar{q}$  process. The algorithms highlight a unique advantage of quantum computations by exploiting a superposition of helicity qubits and allow a simultaneous calculation of the positive and negative helicities of each particle and the  $s$ - and  $t$ -channel amplitudes in the  $2 \rightarrow 2$  process.

Despite recent improvements in the implementation of PS algorithms, running them on NISQ devices and simulating realistic parton showers involving many shower steps are currently challenging. This is largely due to the fact that the shower simulation is performed by repeating many times a circuit corresponding to a single PS step and this often results in a long circuit, which is hard to implement to NISQ device due to limited coherence time, qubit connectivity and hardware noise. One strategy to improve the performance is to mitigate errors through modifications to quantum state operation and measurement protocols. A number of readout and gate error mitigation techniques have been proposed in the literature, e.g., zero-noise extrapolation technique with identity insertions for gate errors, originally proposed in [27] and generalized in [28]. Another complementary strategy to error mitigation is to optimize the quantum circuits in the compilation process. A variety of architecture-agnostic and architecture-specific tools for circuit optimization have been developed, e.g., an industry standard tool called `t|ket>` in [29] from Cambridge Quantum Computing. Ref. [30] introduces a new technique to optimize quantum circuits by identifying the amplitudes of computational basis states and removing redundant controlled gates in polynomial time with quantum measurement. This optimization protocol has been applied to the PS simulation in [24], together with the gate-error mitigation method in [28]. Ref. [30] successfully demonstrates that both the circuit optimization and the error mitigation methods can simplify the circuit significantly and improve the performance on NISQ device, depending on the initial states of the circuit corresponding to different initial particles of parton showers.

### 3 APPLICATIONS OF QUANTUM COMPUTING TO CHARGED PARTICLE PATTERN RECOGNITION

Raw data recorded by detectors is processed by reconstruction algorithms before it can be used for physics analyses. The track reconstruction algorithms used to reconstruct the trajectories of

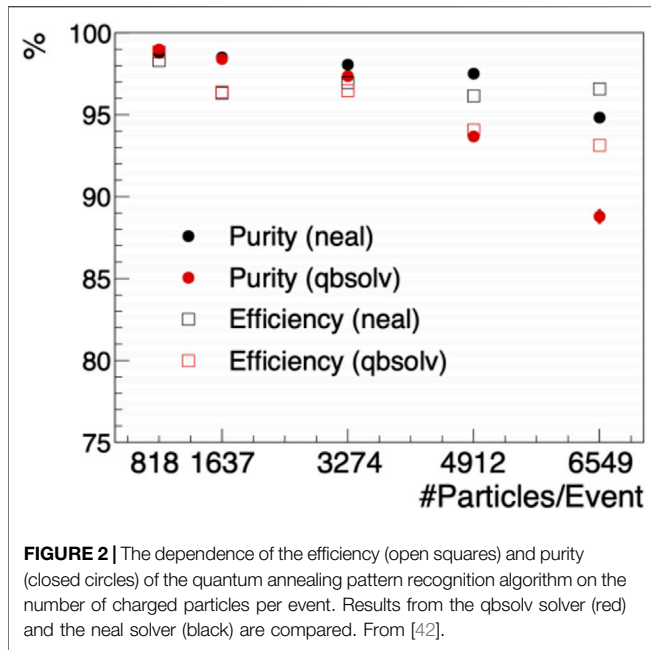
charged particles passing through the tracking detectors are typically the most computationally demanding. The required computing resources for such algorithms scale approximately quadratically with the number of charged particles per event, i.e. the amount of pile up, and therefore will become even more challenging at future colliders. Therefore, new ideas and approaches for track reconstruction algorithms are currently an extremely active field of research to ensure that the physics capabilities of the HL-LHC and beyond can be fully exploited.

Track reconstruction algorithms can be characterized into global and local approaches. Global algorithms process all the data, or hits in the detectors, from an event simultaneously and return a set of tracks. Local algorithms aim to identify the set of hits corresponding to a single track and are run many times to identify the full set of tracks. Examples of global methods include the Hough transform [31, 32] and neural networks [33]. The most widely used local method is the Kalman filter [34–36] and recently there has been extensive exploration into the use of Graph Neural Networks (GNNs) [37]. A number of different track reconstruction algorithms have been explored for quantum computers and, in most cases, the open dataset produced for the tracking machine learning challenge has been used [38, 39]. This dataset will be referred to as the TrackML dataset.

The first track reconstruction algorithm developed for quantum computers is a global track reconstruction algorithm as presented in [40]. The track reconstruction problem is formulated as a QUBO problem and quantum annealers are used to identify the global minimum. This algorithm, as is the case for all quantum algorithms discussed here, should be regarded as a hybrid quantum-classical algorithm because it requires pre- and post-processing on a classical computer.

The algorithm initially groups the hits in the detectors into doublets and then triplets. A QUBO is constructed from the triplets and the goal is to identify which pairs of triplets can be combined to form quadruplets. The weights in the QUBO depend on the compatibility between the properties of the triplets including their curvature and the angles between them, because triplets from the same track are expected to have identical properties. The QUBO is minimized on the quantum annealer by selecting the combinations of triplets compatible with the trajectories of charged particles. However, given the limited number of qubits available on quantum computers today, the QUBO is decomposed into smaller sub-QUBOs that are solved individually using fewer qubits. A software tool, called `qbsolve` [41], from D-Wave is used to perform this splitting and to recombine the solved sub-QUBOs so that the global minimum can be found. After minimization, a final post-processing step is performed on a classical computer to convert the accepted triplets back to doublets. Any duplicates or doublets with unresolved conflicts with other doublets are removed. The final track candidates are required to have at least five hits to reduce the contribution from random combinations of hits, or fakes.

The algorithm was studied using the TrackML dataset but restricting it to the central region of the detector, or barrel, which has a simpler geometry and less material and hence is a simpler problem for pattern recognition algorithms. In addition, events were filtered to select particular fractions of particles to emulate



datasets with different amounts of pile up. This allows the dependence of the performance on the amount of pile up to be studied. The performance was studied using simulations of quantum annealers on Cori, a supercomputer located at the National Energy Research Scientific Computing Center (NERSC), and on quantum annealing hardware from D-Wave. Two different quantum annealers were used: the Ising D-Wave 2X located at Los Alamos National Laboratory and the D-Wave LEAP cloud service, which is an interface provided by D-Wave that allows users to run on a number of different quantum computers.

Key performance metrics for track reconstruction include the efficiency and the purity. The efficiency is the fraction of true particles that are successfully reconstructed and the purity is the fraction of reconstructed tracks that correspond to true particles. Ref. [40] showed that the efficiency is 100% in events with low pile up and it decreased to 90% in events with the level of pile up expected at the HL-LHC. However, while the purity is close to 100% at low pile up, it decreases rapidly with increasing pile up to reach only 50% at HL-LHC multiplicities. This demonstrates that the algorithm is impacted by fake tracks from random combinations of hits. The performance on the quantum hardware and in simulation is found to be consistent, which demonstrates that the algorithm is not significantly impacted by the noise on the quantum annealer.

Ref. [42] improves the performance of this quantum annealing pattern recognition algorithm by modifying the weights in the QUBO to include information about the impact parameters of the triplets. The impact parameters provide a measure of the distance of closest approach of tracks to the location where the proton-proton collision occurred. They are used by [42] to preferentially select tracks produced at the primary interaction point over secondaries produced in decays of primary particles or

through interactions in the detector material. The efficiency and purity are shown in **Figure 2** using simulation as a function of the number of particles per event and results are shown for two different solving algorithms: qbsolv and Neal. While the efficiency is slightly improved over the results shown in [40], the purity is dramatically improved to 85% or 95% depending on the solver at HL-LHC multiplicities. While this approach is very effective at reducing the fake rate, it is also expected to have a low efficiency to reconstruct tracks from secondary decays, such as  $B$ -hadrons and  $\tau$ -leptons. The performance of Neal is superior to qbsolv in all cases.

[42] also studied the performance of annealing algorithms for pattern recognition using a digital annealing machine from Fujitsu. This is a quantum-inspired classical computer specifically formulated to solve annealing problems. Comparable physics performance was obtained between the quantum and digital annealers, however the computational time on the digital annealer was found to be far superior to the quantum annealer and to be essentially independent of the amount of pile up in the event.

A similar algorithm for pattern recognition on quantum annealers is presented in [43]. They also used triplets and the objective function depends on the angles between the triplets, a bias term to preferentially select high-momentum tracks and the point of origin of the tracks. There are penalty terms corresponding to bent and poorly oriented tracks.

The TrackML dataset was also used and tracks were reconstructed in both the barrel and the endcap of the detector. To ensure that the problem could be solved on quantum annealers available today, the dataset was split geometrically into 32 sectors and then sub-QUBOs were defined within each sector. Results were obtained both using simulation and a D-Wave 2X machine. In most regions of phase space, the efficiency was found to be approximately 90% and the purity greater than 95%.

Track reconstruction algorithms are also used in trigger detectors to select events which are subsequently processed in more detail for offline reconstruction. For such applications, algorithms need to be run in real time but can tolerate lower precision than offline reconstruction. One approach, which has been considered is the use of a memory bank of patterns of measurements. This allows track reconstruction algorithms to be replaced by looking up patterns in the memory bank. This means that the problem of the amount of processing power needed is transformed into memory requirements.

[44] explored how quantum computers, with their potential for exponential storage capacity, can be used for such problems. While the amount of classical memory required to store patterns depends exponentially on the number of elementary memory units, the amount of quantum memory depends only logarithmically, which means that far less memory is required. To demonstrate this, they presented an implementation of a quantum associative memory protocol (QuAM) in [45] for 2-bit patterns circuit-based quantum computers using the IBM Qiskit framework. They used Trugenberger's algorithm to store the patterns and the generalized Grover's algorithm to retrieve the patterns.

They explored IBM circuit-based quantum computers with up to 14 qubits, however, these did not provide sufficient qubits to solve 2-bit patterns.

An additional approach for charged particle pattern recognition, relying on quantum machine learning, a Quantum Graph Neural Network (QGNN) will be discussed in **Section 4.2**.

## 4 APPLICATIONS OF QUANTUM MACHINE LEARNING

Recent developments of quantum computing architecture and algorithms make Quantum Machine Learning (QML) a promising early application in NISQ era. QML includes a wide range of research topics, e.g., information theory aspects such as quantum learning complexity, accuracy and asymptotic behavior in a fault-tolerant regime, as well as more near-term aspects such as data encoding, learning circuit models and hybrid architectures with classical calculations. An important aspect of QML for early HEP applications [46] is how effectively one can exploit machine learning to characterize a quantum state generated from input data in a quantum-classical combined setting. The quantum-classical hybrid approach is particularly useful for near-term quantum devices because the quantum part can focus on classically-intractable calculations while the classical part handles the rest of the computation including, e.g., optimization of the learning model. Classical machine learning is used extensively in the analysis of HEP experimental data. The significant growth of the data volume foreseeable for the HL-LHC era and beyond further motivates quantum-empowered machine learning in the workflow of future HEP experiments.

A number of experimental applications of QML to data analysis and event reconstruction in representative HEP experiments have been explored. They are categorized into two approaches based on quantum annealing and quantum circuit models, each having its own advantages and disadvantages in terms of the maturity and applicability. Most QML applications in HEP (discussed below) belong to supervised machine learning, but several applications (mentioned explicitly) use unsupervised learning techniques.

### 4.1 Quantum Machine Learning With Quantum Annealing

QA-based machine learning (QAML) formulates ML as a QUBO problem (see **Section 3**) and looks for the best classification. This is done by minimizing the cost function that quantifies the difference between the predictions and the true labels. In [47], QAML is applied to the classification of Higgs events where the Higgs boson decays into two photons, one of the key channels used for the Higgs discovery. From eight high-level kinematic features in a set of training events  $\{\mathbf{x}_\tau, \mathbf{y}_\tau\}$  with  $\tau$  being the event index, 36 *weak* classifiers  $c_i(\mathbf{x}_\tau)$  are constructed (with  $i$  being the classifier index) in such a way that the signal (background) events populate at positive (negative) values within the range between  $-1$

and 1. The term *weak* indicates that these classifiers are constructed from various combinations of arithmetic operations on the kinematic features. A *strong* classifier is then constructed as a linear combination of the weak classifiers with binary coefficients  $w_i \in \{0, 1\}$ . The objective function is defined as:

$$O(\mathbf{w}) = \sum_{i,j} C_{ij} w_i w_j + \sum_i (\lambda - 2C_i) w_i, \quad (1)$$

where  $C_{ij} = \sum_\tau c_i(\mathbf{x}_\tau) c_j(\mathbf{x}_\tau)$ ,  $C_i = \sum_\tau c_i(\mathbf{x}_\tau) \mathbf{y}_\tau$  with the true label  $\mathbf{y}_\tau \in \{+1, -1\}$ , and  $\lambda$  is a positive parameter to penalize the number of non-zero weight,  $w_i$ , terms. The objective function  $O$  is finally transformed into the problem Hamiltonian by converting the binary  $w_i$  to spin variable  $\sigma_z^i \in \{-1, 1\}$  for quantum annealing.

QAML performance for Higgs classification in two-photon events is compared in [47] with two classical ML techniques based on Boosted Decision Tree (BDT) and Deep Neural Network (DNN). In addition, the same problem Hamiltonian is adopted to Simulated Annealing (SA), a classical analogue of QA with energy fluctuations controlled by artificial temperature variables. The results show that the QAML and SA have similar classification performance in terms of the area under receiver operating characteristic (ROC) curve for the true-positive and false-positive rates, which correspond to the signal and background efficiencies, respectively. The annealing-based methods have no clear advantage over the classical ML methods, though a hint of slight advantage is seen at small dataset size. The QAML method has been further extended in [48] to the so-called QAML-Z, that aims to zoom into the energy surface to optimize real-valued coefficients and sequentially apply QA to an augmented set of weak classifiers. The QAML-Z method is applied in [49] to investigate the selection of Supersymmetric top (stop) quark events against standard model background events.

### 4.2 Quantum Machine Learning With Quantum Circuits

There are two widely-used ML implementations for near-term gate-based quantum computers: Variational Quantum Algorithms (VQA) in [50] and Quantum Support Vector Machines (QSVM) with kernel method in e.g., [51] and [52]. For both approaches, the input classical data  $\mathbf{x} \in \mathbb{R}^d$  is embedded in a Hilbert space with a unitary operator  $U(\mathbf{x}_i)$  to create a quantum state  $|\phi(\mathbf{x}_i)\rangle = U(\mathbf{x}_i)|0\rangle^{\otimes n}$ , where  $n$  is the number of qubits. Since the NISQ devices have only limited resources, simplified problems have been considered for both approaches with the circuit-based machine learning.

VQA uses an ansatz created using unitary operator  $V(\boldsymbol{\theta})$  with tunable parameters  $\boldsymbol{\theta}$  and measures the produced final state  $V(\boldsymbol{\theta})|\phi(\mathbf{x}_i)\rangle$  with certain Hermitian operators. A cost function is defined from the measurement outcome and the input ground truth, and the parameter optimization or *training* is performed classically by minimizing the cost function. The minimization of cost function is an important subject in the field, and one of the representative methods exploits the gradients of the cost

function with respect to the  $\theta$  parameters, which is so-called the parameter-shift rule [53–56]. This VQA approach has a wide range of applications including classification, regression and optimization.

QSVM performs classification in high-dimensional feature space by constructing the kernel function from the inner products of embedded quantum states  $K(x_i, x_j) = |\langle \phi(x_i) | \phi(x_j) \rangle|^2$ . The kernel function is sampled by estimating the probability of measuring the  $|0\rangle^{\otimes n}$  state from the circuit and is used by the classical SVM to build a hyperplane for data classification. QSVM with a kernel function often becomes more efficient than VQA in the training stage because the parameter tuning is performed fully classically with the obtained kernel function for the SVM. A polynomial number of quantum gates used in VQA and QSVM, acting on a quantum state that spans a Hilbert space exponentially, could bring quantum advantage with near-term NISQ devices in the future.

VQA-based HEP applications have largely focused on the classification of physics process of interests so far. They include the classifications of Higgs boson [57, 58], supersymmetric particles with missing transverse momentum [59], neutrino events [60], new resonances in di-top final states [61] and  $B\bar{B}$  production in  $e^+e^-$  collisions [62].

[57] studies the classification of  $t\bar{t}H$  events with  $H \rightarrow \gamma\gamma$  decays and  $H \rightarrow \mu^+\mu^-$  events from the dominant background processes for each channel using VQA. After reducing the kinematic variables into 10 variables using a principal component analysis, [57] encodes the input data as angles of  $R_Z$  rotation gates, and produces an ansatz composed of  $R_Y$  and  $R_Z$  rotation gates with parameters and an entangling layer of controlled-Z gates between adjacent qubits. Binary classification is performed with the VQA framework using both simulation and quantum hardware. The results show that the VQA approach has a discrimination power similar to the classical SVM and BDT, and the performance with quantum device is comparable to that of a quantum simulator.

Ref. [59] explores the classification of the production of SUSY events with two  $W \rightarrow l\nu$  decays and two neutralinos against a SM background process of two  $W \rightarrow l\nu$  production. Restricting the input features to 3, 5 and 7 kinematic variables, two different implementations of VQA with different configurations of data encoding and ansatz circuits are investigated using a quantum simulator and quantum hardware. As either the number of input variables or training events in the sample increase, the classical techniques using BDTs and DNNs outperform the VQA classifiers in the simulator. However, the VQA classifiers have comparable performance when the training sample size or the number of input variables is small.

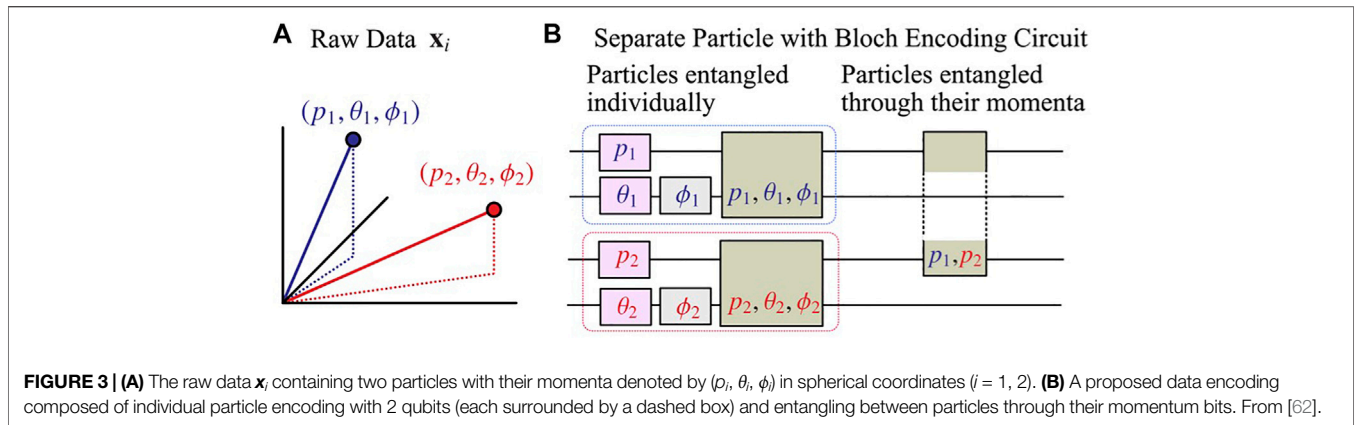
In general, it is interesting to investigate the impact of data encoding on VQA performance, or more specifically if there is any efficient way to encode data from HEP experiments. [62] attempts to address this question in the classification of  $e^+e^- \rightarrow \Upsilon(4S) \rightarrow B\bar{B}$  events with  $B \rightarrow K^+K^-$  decays from  $e^+e^- \rightarrow q\bar{q}$  background events. The momenta of the final state particles (input data  $\mathbf{x}$ ) are encoded into the quantum states  $|\psi(\mathbf{x})\rangle$ . The kernel  $K(\mathbf{x}_i, \mathbf{x}_j) = |\langle \psi(\mathbf{x}_i) | \psi(\mathbf{x}_j) \rangle|^2$  is derived by measuring the output state, which feeds into the classical

SVM for the classification. In the encoding stage, input particle momenta are encoded by maintaining the kinematic properties of individual particles with a dense-encoding technique. An example data encoding gate proposed by the authors is shown in **Figure 3**. In this encoding, individual particles are encoded first with their momentum variables ( $p, \theta, \phi$ ) in spherical coordinates, then each particle is entangled with the other in the event through one of their qubits. [62] demonstrates that the classification performance is improved by exploiting this encoding scheme for the classification of  $e^+e^- \rightarrow \Upsilon(4S) \rightarrow B\bar{B}$  events.

Ref. [58] considers the classification of  $t\bar{t}H$  with  $H \rightarrow b\bar{b}$  decays using 67 input features. The authors first perform feature reduction to adapt the large number of input features to the capacity of current quantum computation with techniques based on auto-encoding and feature selection. The VQA and QSVM performance are compared with a DNN and a BDT. The Area Under Curve (AUC) values of the quantum classifiers are similar to those from the DNN and BDT, indicating that the performance is similar.

In the absence of a clear new physics signal in HEP experiments, a data-driven, signal-agnostic search has gained considerable interests over years. Unsupervised machine learning for new physics searches, often realized as anomaly detection, will be an important tool in future colliders. Several pioneering studies with unsupervised QML technique have been done in the contexts of anomaly detection. Ref. [63] attempts to combine an anomaly detection technique with a graph representation of HEP events through quantum computation. In particular, the continuous-variable (CV) model of quantum computer, programmed using photonic quantum device, is exploited to survey graph-represented events to search for  $pp \rightarrow HZ$  signal. Ref. [63] uses Gaussian boson sampling of CV events as input into a quantum variant of K-means clustering algorithm for anomaly detection. A discrete (qubit-based) QML approach for unsupervised learning is also investigated in [64], where the feasibility of anomaly detection is explored using quantum autoencoders (QAE) based on variational quantum circuits. With the benchmark process of  $pp \rightarrow H \rightarrow t\bar{t}$  for signal, the QAE performance has been compared with that from a classical autoencoder for anomaly detection.

Various classical ML techniques have been applied to the tasks of detector simulation and reconstruction in HEP experiments (see the review in HEP ML Community). Among them, the Generative Adversarial Network (GAN) is extensively studied to simulate calorimeter energy deposits of particle showers as images, aiming for increased speed compared to a full Geant4-based detector simulation. The quantum version of a GAN (QGAN) and its variants have been proposed and investigated for HEP detector simulation [65] and Monte Carlo event generation [66]. Ref. [67] employs the Parameterized Quantum Circuit (PQC) GAN model, which is composed of a quantum Generator and a classical Discriminator to demonstrate a proof-of-concept for the QGAN-based shower simulation. In particular, the GAN model in [67] uses two PQCs to sample two probability distributions, one for the shower images and the other for normalized pixel intensities in a single image. This allows a



full training set of images to be captured. Even though the image size is restricted only to  $2 \times 2$  pixels, the proposed Dual-PQC GAN model manages to generate a sample of individual images and their probability distributions consistent with those in the training set.

A technique from machine learning that has been applied to the problem of charged particle pattern recognition are GNNs. Ref. [68, 69] explored how GNNs can be applied to the problem of pattern recognition. They developed a hybrid quantum-classical algorithm, called a Quantum GNN (QGNN) that relies on a series of quantum edge and quantum node networks. After iterating through the series, a final edge network obtains the final segment classification. [69] showed that the QGNN performs similarly to classical methods when the number of hidden dimensions is low. They studied the scaling performance of the QGNN in simulation by varying the number of hidden dimensions and qubits. The QGNN performance improves with additional hidden dimensions, however, some saturation of the best loss is observed. The results were limited to simulation due to the current status of NISQ hardware. The application of this technique to the LUXE experiment is explored in [70].

## 5 CONCLUSION AND OUTLOOK

Quantum computing has been undergoing explosive progress recently and may have the potential to provide solutions to computing challenges in HEP. There are many potential applications and we have reviewed algorithms for quantum simulation, quantum pattern recognition and quantum machine learning and their challenges.

QFT simulation with quantum computers is highly motivated because universal quantum computers can perform such simulation with exponentially smaller computing resources than classical computers. However, it would require many more physical qubits than are currently available. Several representative studies focusing on a simplified QFT, in particular parton showers and hard scatterings, have been reviewed here. These studies considered simple cases compared to the realistic MC simulations currently employed in HEP experiments, but demonstrate that interesting quantum

properties can be captured even with current devices. A full QFT simulation envisioned by Jordan, Lee and Preskill will certainly require many logical qubits free from errors, composed of millions of physical qubits, called Fault-Tolerant Quantum Computers (FTQC). The realization of FTQC is however still decades away. A near-term goal of quantum simulation is to develop quantum algorithms for each QFT simulation step and evaluate the feasibility to realize potential advantage over classical simulation with near-term technologies.

Pattern recognition algorithms are among the most computationally demanding algorithms for HEP. A number of studies have focused on global algorithms using quantum annealers. These are typically hybrid quantum-classical algorithms with pre- and post-processing performed on classical computers and finding the solution on a quantum annealer. All algorithms were able to obtain excellent efficiencies, however some reported significant rates of fake tracks particularly in very busy events. In all cases, the problems needed to be simplified to be able to run on NISQ devices. A study on the use of quantum circuits as associated memory for triggers was presented, which would exploit the exponential storage capacity of quantum computers. In the future, it would be interesting to explore local pattern recognition algorithms, particularly on circuit-based computers. One of the main challenges for such algorithms is that they require large amounts of data to be processed on the quantum device, and the transfer of this data may limit the speed of such algorithms.

Quantum machine learning is a promising early application in the NISQ era, not only for HEP but also for other scientific domains and industries. Significant progress has been made over the past few years in aspects including data encoding, ansatz design and trainability for VQA and kernel-based learning architecture. The QAML technique has been employed for the Higgs-boson classification and demonstrated that the classification performance is comparable to the classical approaches based on BDT and DNN. A number of VQA- and QSVM-based classification studies have been demonstrated in representative data analyses for  $pp$  and  $e^+e^-$  collisions. The quantum classifiers have comparable performance to the conventional BDT and/or DNN methods, meaning that quantum advantage has not



been demonstrated yet. Anomaly detection using unsupervised QML methods is being explored. The QML approach has also been investigated in the detector simulation as well, with the demonstration of reproducing a calorimeter shower of energy deposits in a simplified setting. QML applications to HEP data analysis will pose several challenges in the future. A large-volume, high-dimensional data produced from HEP experiments needs to be embedded into quantum states, which requires significant computing resources. The encoded states are then processed in VQA by a parameterized quantum circuit to extract underlying properties in the data. Recent studies show that the ansatz needs to have sufficient expressibility for a given problem but should not be too expressive, making it exponentially harder to compute the gradients of cost function with the number of qubits. This infamous problem, so-called barren plateau in the cost function landscape, and the data encoding are currently very active areas of research for QML. It will be crucial to explore them further in the context of HEP-oriented QML

architectures towards realizing quantum advantage for QML in HEP.

## AUTHOR CONTRIBUTIONS

The main contributions from HG were **sections 1 and 3** and the main contributions from KT were **sections 2 and 4**.

## FUNDING

HG has received funding from the Alfred P. Sloan Foundation under grant no. FG-2020-13732. KT has received funding from the U.S.-Japan Science and Technology Cooperation Program in High Energy Physics under the project “Optimization of HEP Quantum Algorithms” (FY2021-23).

## REFERENCES

- Berners-Lee TJ, Cailliau R, Groff JF, Pollermann B. World Wide Web: An Information Infrastructure for High-Energy Physics. *Conf Proc C* (1992) 920:157–64.
- Foster I, Kesselman C, Tuecke S. The Anatomy of the Grid - Enabling Scalable Virtual Organizations. *The International Journal of High Performance Computing Applications* (2001). 15 3:200–222. doi:10.1177/109434200101500302
- Bruening OS, Collier P, Lebrun P, Myers S, Ostojic R, Poole J, et al. *LHC Design Report. Tech. Rep. CERN-2004-003-V-1*. Geneva: CERN Yellow Reports: Monographs (2004). doi:10.5170/CERN-2004-003-V-1
- Apollinari G, Béjar Alonso I, Brüning O, Fessia P, Lamont M, Rossi L, et al. *High-Luminosity Large Hadron Collider (HL-LHC): Technical Design Report V. 0.1*. Geneva: Tech. Rep. CERN Yellow Reports: Monographs (2017).
- ATLAS Collaboration. *ATLAS Software and Computing HL-LHC Roadmap. Tech. Rep.* Geneva: CERN (2022).
- Software CO Computing. *Evolution of the CMS Computing Model towards Phase-2Tech. Rep.* Geneva: CERN (2021).
- Shadura O, Krikler B, Stewart GA. HL-LHC Computing Review Stage 2, Common Software Projects: Data Science Tools for Analysis. In: J Pivarski, E Rodrigues, K Pedro, editors. *LHCC Review of HL-LHC Computing* (2022).
- Yazgan E. *HL-LHC Computing Review Stage-2. Common Software Projects: Event Generators* (2021). doi:10.48550/arXiv.2109.14938
- Abada A. FCC-hh: The Hadron Collider: Future Circular Collider Conceptual Design Report Volume 3. *Eur Phys J ST* (2019) 228:755–1107. doi:10.1140/epjst/e2019-900087-0
- Preskill J. Quantum Computing in the NISQ Era and beyond. *Quantum* (2018) 2:79. doi:10.22331/q-2018-08-06-79
- Bruzewicz CD, Chiaverini J, McConnell R, Sage JM. Trapped-ion Quantum Computing: Progress and Challenges. *Appl Phys Rev* (2019) 6:021314. doi:10.1063/1.5088164
- Kjaergaard M, Schwartz ME, Braumüller J, Krantz P, Wang JI-J, Gustavsson S, et al. Superconducting Qubits: Current State of Play. *Annu Rev Condens Matter Phys* (2020) 11:369–95. doi:10.1146/annurev-conmatphys-031119-050605
- Preskill J. *Quantum Computing 40 Years Later* (2021). doi:10.48550/arXiv.2106.10522
- [Dataset] HEP ML Community. *A Living Review of Machine Learning for Particle Physics* (2021). doi:10.48550/arXiv.2102.02770
- Jordan SP, Lee KSM, Preskill J. Quantum Algorithms for Quantum Field Theories. *Science* (2012) 336:1130–3. doi:10.1126/science.1217069
- Preskill J. *Simulating Quantum Field Theory with a Quantum Computer* (2018). doi:10.48550/arXiv.1811.10085
- Argüelles CA, Jones BJP. Neutrino Oscillations in a Quantum Processor. *Phys Rev Res* (2019) 1:033176. doi:10.1103/physrevresearch.1.033176
- Jha AK, Chatla A, Bamba BA. *Quantum Simulation of Oscillating Neutrinos* (2020). doi:10.48550/arXiv.2010.06458
- Chang SY. *Quantum Generative Adversarial Networks in a Continuous-Variable Architecture to Simulate High Energy Physics Detectors* (2021). doi:10.48550/arXiv.2101.11132
- de Jong WA, Metcalf M, Mulligan J, Płoskoń M, Ringer F, Yao X. Quantum Simulation of Open Quantum Systems in Heavy-Ion Collisions. *Phys Rev D* (2021) 104:L051501. doi:10.1103/physrevd.104.l051501
- Pérez-Salinas A, Cruz-Martinez J, Alhajri AA, Carrazza S. Determining the Proton Content with a Quantum Computer. *Phys Rev D* (2021) 103. doi:10.1103/physrevd.103.034027
- Bauer CW, Nachman B, Freytsis M. Simulating Collider Physics on Quantum Computers Using Effective Field Theories. *Phys Rev Lett* (2021) 127. doi:10.1103/physrevlett.127.212001
- Stetina TF, Ciavarella A, Li X, Wiebe N. Simulating Effective QED on Quantum Computers. *Quantum* (2022) 6:622. doi:10.22331/q-2022-01-18-622
- Nachman B, Provasoli D, de Jong WA, Bauer CW. Quantum Algorithm for High Energy Physics Simulations. *Phys Rev Lett* (2021) 126:062001. doi:10.1103/PhysRevLett.126.062001
- Williams S, Malik S, Spannowsky M, Bepari K. *A Quantum Walk Approach to Simulating Parton Showers* (2021). doi:10.48550/arXiv.2109.13975
- Bepari K, Malik S, Spannowsky M, Williams S. Towards a Quantum Computing Algorithm for Helicity Amplitudes and Parton Showers. *Phys Rev D* (2021) 103:076020. doi:10.1103/PhysRevD.103.076020
- Endo S, Benjamin SC, Li Y. Practical Quantum Error Mitigation for Near-Future Applications. *Phys Rev X* (2018) 8:031027. doi:10.1103/PhysRevX.8.031027
- He A, Nachman B, de Jong WA, Bauer CW. Zero-noise Extrapolation for Quantum-Gate Error Mitigation with Identity Insertions. *Phys Rev A* (2020) 102:012426. doi:10.1103/physreva.102.012426
- Sivarajah S, Dilkes S, Cowtan A, Simmons W, Edgington A, Duncan R.  $t|ket\rangle$ : a Retargetable Compiler for NISQ Devices. *Quan Sci. Technol.* (2020) 6:014003. doi:10.1088/2058-9565/ab8e92
- Jang W, Terashi K, Saito M, Bauer CW, Nachman B, Iiyama Y, et al. Quantum Gate Pattern Recognition and Circuit Optimization for Scientific Applications. *EPJ Web Conf* (2021) 251:03023. doi:10.1051/epjconf/202125103023

31. Hough P. Machine Analysis of Bubble Chamber Pictures. *Conf Proc C* (1959) 590914:554–8. Available at: <https://cds.cern.ch/record/107795?ln=en>
  32. Duda RO, Hart PE. Use of the Hough Transformation to Detect Lines and Curves in Pictures. *Commun ACM* (1972) 15:11–5. doi:10.1145/361237.361242
  33. Hopfield JJ. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. *Proc Natl Acad Sci U.S.A* (1982) 79:2554–8. doi:10.1073/pnas.79.8.2554
  34. Kalman RE. A New Approach to Linear Filtering and Prediction Problems. *J Basic Eng* (1960) 82:35–45. doi:10.1115/1.3662552
  35. Billoir P. Track Fitting with Multiple Scattering: A New Method. *Nucl Instr Methods Phys Res* (1984) 225:352–66. doi:10.1016/0167-5087(84)90274-6
  36. Frühwirth R. Application of Kalman Filtering to Track and Vertex Fitting. *Nucl Instr Methods Phys Res Section A: Acc Spectrometers, Detectors Associated Equipment* (1987) 262:444–50. doi:10.1016/0168-9002(87)90887-4
  37. Duarte J, Vlimant J-R. Graph Neural Networks for Particle Tracking and Reconstruction. *arxiv* (2022) 387–436. doi:10.1142/9789811234033\_0012
  38. Amrouche S, Basara L, Calafiura P, Estrade V, Farrell S, Ferreira DR, et al. The Tracking Machine Learning challenge: Accuracy Phase. *The NeurIPS* (2020) 18:231–64. doi:10.1007/978-3-030-29135-8\_9
  39. Amrouche S. and others. *The Tracking Machine Learning challenge : Throughput Phase* (2021). doi:10.48550/arXiv.2105.01160
  40. Bapst F, Bhimji W, Calafiura P, Gray H, Lavrijsen W, Linder L, et al. A Pattern Recognition Algorithm for Quantum Annealers. *Comput Softw Big Sci* (2020) 4:1. doi:10.1007/s41781-019-0032-5
  41. Booth M, Reinhardt S, Roy A. *Partitioning Optimization Problems for Hybrid Classical/quantum Execution*. Tech. Rep. dWave Technical Report (2017). Available at: [https://docs.ocean.dwavesys.com/projects/qbsolv/en/latest/\\_downloads/bd15a2d8f32e587e9e5997ce9d5512cc/qbsolv\\_techReport.pdf](https://docs.ocean.dwavesys.com/projects/qbsolv/en/latest/_downloads/bd15a2d8f32e587e9e5997ce9d5512cc/qbsolv_techReport.pdf)
  42. Saito M, Calafiura P, Gray H, Lavrijsen W, Linder L, Okumura Y, et al. Quantum Annealing Algorithms for Track Pattern Recognition. *EPJ Web Conf* (2020) 245:10006. doi:10.1051/epjconf/202024510006
  43. Zlokapa A, Anand A, Vlimant JR, Duarte JM, Job J, Lidar D, et al. *Charged Particle Tracking with Quantum Annealing-Inspired Optimization* (2019). doi:10.48550/arXiv.1908.04475
  44. Shapoval I, Calafiura P. Quantum Associative Memory in HEP Track Pattern Recognition. *EPJ Web Conf* (2019) 214:01012. doi:10.1051/epjconf/201921401012
  45. Ventura D, Martinez T. Quantum Associative Memory with Exponential Capacity. 1998 IEEE International Joint Conference on Neural Networks Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98CH36227) (1998) 11:509–13. doi:10.1109/IJCNN.1998.682319
  46. Guan W, Perdue G, Pesah A, Schuld M, Terashi K, Vallecorsa S, et al. Quantum Machine Learning in High Energy Physics. *Mach Learn Sci Technol* (2021) 2:011003. doi:10.1088/2632-2153/abc17d
  47. Mott A, Job J, Vlimant J-R, Lidar D, Spiropulu M. Solving a Higgs Optimization Problem with Quantum Annealing for Machine Learning. *Nature* (2017) 550:375–9. doi:10.1038/nature24047
  48. Zlokapa A, Mott A, Job J, Vlimant J-R, Lidar D, Spiropulu M. Quantum Adiabatic Machine Learning by Zooming into a Region of the Energy Surface. *Phys Rev A* (2020) 102:062405. doi:10.1103/PhysRevA.102.062405
  49. Bargassa P, Cabos T, Cordeiro Oudot Choi AA, Hessel T, Cavinato S. Quantum Algorithm for the Classification of Supersymmetric Top Quark Events. *Phys Rev D* (2021) 104:096004. doi:10.1103/PhysRevD.104.096004
  50. Cerezo M, Arrasmith A, Babbush R, Benjamin SC, Endo S, Fujii K, et al. Variational Quantum Algorithms. *Nat Rev Phys* (2021) 3:625–44. doi:10.1038/s42254-021-00348-9
  51. Havlíček V, Córcoles AD, Temme K, Harrow AW, Kandala A, Chow JM, et al. Supervised Learning with Quantum-Enhanced Feature Spaces. *Nature* (2019) 567:209–12. doi:10.1038/s41586-019-0980-2
  52. Schuld M, Killoran N. Quantum Machine Learning in Feature Hilbert Spaces. *Phys Rev Lett* (2019) 122:040504. doi:10.1103/PhysRevLett.122.040504
  53. Gian Giacomo Guerreschi MS. *Practical Optimization for Hybrid Quantum-Classical Algorithms* (2017). doi:10.48550/arXiv.1701.01450
  54. Mitarai K, Negoro M, Kitagawa M, Fujii K. Quantum Circuit Learning. *Phys Rev A* (2018) 98:032309. doi:10.1103/PhysRevA.98.032309
  55. Schuld M, Bergholm V, Gogolin C, Izaac J, Killoran N. Evaluating Analytic Gradients on Quantum Hardware. *Phys Rev A* (2019) 99:032331. doi:10.1103/PhysRevA.99.032331
  56. Mari A, Bromley TR, Killoran N. Estimating the Gradient and Higher-Order Derivatives on Quantum Hardware. *Phys Rev A* (2021) 103:012405. doi:10.1103/PhysRevA.103.012405
  57. Wu SL, Chan J, Guan W, Sun S, Wang A, Zhou C, et al. Application of Quantum Machine Learning Using the Quantum Variational Classifier Method to High Energy Physics Analysis at the LHC on IBM Quantum Computer Simulator and Hardware with 10 Qubits. *J Phys G: Nucl Part Phys* (2021) 48:125003. doi:10.1088/1361-6471/ac1391
  58. Belis V, González-Castillo S, Reissel C, Vallecorsa S, Combarro EF, Dissertori G, et al. Higgs Analysis with Quantum Classifiers. *EPJ Web Conf* (2021) 251:03070. doi:10.1051/epjconf/202125103070
  59. Terashi K, Kaneda M, Kishimoto T, Saito M, Sawada R, Tanaka J. Event Classification with Quantum Machine Learning in High-Energy Physics. *Comput Softw Big Sci* (2021) 5:2. doi:10.1007/s41781-020-00047-7
  60. Chen SYC. Quantum Convolutional Neural Networks for High Energy Physics Data Analysis. *Phys. Rev. Res.* 4, 013231. doi:10.1103/PhysRevResearch.4.013231 (2020).
  61. Blance A, Spannowsky M. Quantum Machine Learning for Particle Physics Using a Variational Quantum Classifier. *J High Energy Phys* (2021) 2021:212. doi:10.1007/jhep02(2021)212
  62. Heredge J, Hill C, Hollenberg L, Sevier M. *Quantum Support Vector Machines for Continuum Suppression in B Meson Decays* (2021). doi:10.48550/arXiv.2103.12257
  63. Blance A, Spannowsky M. Unsupervised Event Classification with Graphs on Classical and Photonic Quantum Computers. *J High Energy Phys* (2021) 2021. doi:10.1007/jhep08(2021)170
  64. Ngairangbam VS, Spannowsky M, Takeuchi M. *Anomaly Detection in High-Energy Physics Using a Quantum Autoencoder* (2021). doi:10.48550/arXiv.2112.04958
  65. Chang SY, Herbert S, Vallecorsa S, Combarro EF, Duncan R. Dual-parameterized Quantum Circuit gan Model in High Energy Physics. *EPJ Web Conf* (2021) 251:03050. doi:10.1051/epjconf/202125103050
  66. Bravo-Prieto C. *Style-based Quantum Generative Adversarial Networks for Monte Carlo Events* (2021). doi:10.48550/arXiv.2110.06933
  67. Ying C, Yunheng SZ. *Quantum Simulations of the Non-unitary Time Evolution and Applications to Neutral-Kaon Oscillations* (2021).
  68. Tüysüz C, Carminati F, Demirköz B, Dobos D, Fracas F, Novotny K, et al. A Quantum Graph Neural Network Approach to Particle Track Reconstruction (2020). doi:10.5281/zenodo.4088474
  69. Tüysüz C, Rieger C, Novotny K, Demirköz B, Dobos D, Potamianos K, et al. Hybrid Quantum Classical Graph Neural Networks for Particle Track Reconstruction. *Quan Mach. Intell.* (2021) 3:29. doi:10.1007/s42484-021-00055-9
  70. Funcke L, Hartung T, Heinemann B, Jansen K, Kropf A, Kühn S. Studying Quantum Algorithms for Particle Track Reconstruction in the LUXE experiment. In: 20th International Workshop on Advanced Computing and Analysis Techniques in Physics Research: AI Decoded - Towards Sustainable, Diverse, Performant and Effective Scientific Computing (2022).
- Conflict of Interest:** The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.
- The reviewer GNP declared a past collaboration with one of the authors KT to the handling editor.
- Publisher's Note:** All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.
- Copyright © 2022 Gray and Terashi. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.