PROCEEDINGS OF SCIENCE

PoS(EPS-HEP2021)842

Application of Quantum Machine Learning to HEP Analysis at LHC using Quantum Computer Simulators and Quantum Computer Hardware

S. L. Wu*,[a] **J. Chan,**[a] **A. Cheng,**[a] **W. Guan,**[a] **S. J. Sun,**[a] **R. Zhang,**[a] **C. Zhou,**[a] **M. Livny,**[b] **A. D. Meglio,**[c] **A. C. Y. Li,**[d] **J. Lykken,**[d] **P. Spentzouris,**[d] **S. Y.-C. Chen,**[e] **S. Yoo,**[e] **T.-C. Wei,**[f] **P. Lougovski,**[g] **S. Padhi,** [g] **S. Severini,**[g] **and D. Walker**[g]

[a]*Physics Department of University of Wisconsin-Madison, US*[†]
[b]*Computer Sciences Department of University of Wisconsin-Madison, US*
[c]*CERN Quantum Technology Initiative, CH*
[d]*Quantum Institute, Fermilab, US*
[e]*Computational Science Initiative, BNL, US*
[f]*C.N. Yang Institute for Theoretical Physics, State University of New York at Stony Brook, US*
[g]*Quantum Computing and AI Research, Amazon Web Services, US*

E-mail: sau.lan.wu@cern.ch

© Copyright owned by the author(s) under the terms of the Creative Commons
Attribution-NonCommercial-NoDerivatives 4.0 International License (CC BY-NC-ND 4.0).

Machine learning enjoys widespread success in High Energy Physics (HEP) analyses at LHC. However the ambitious HL-LHC program will require much more computing resources in the next two decades. Quantum computing may offer speed-up for HEP physics analyses at HL-LHC, and can be a new computational paradigm for big data analyses in High Energy Physics.

We have successfully employed three methods (1) Variational Quantum Classifier (VQC) method, (2) Quantum Support Vector Machine Kernel (QSVM-kernel) method and (3) Quantum Neural Network (QNN) method for two LHC flagship analyses: ttH (Higgs production in association with two top quarks) and H->mumu (Higgs decay to two muons, the second generation fermions). We shall address the progressive improvements in performance from method (1) to method (3).

We will present our experiences and results of a study on LHC High Energy Physics data analyses with IBM Quantum Simulator and Quantum Hardware (using IBM Qiskit framework), Google Quantum Simulator (using Google Cirq framework), and Amazon Quantum Simulator (using Amazon Braket cloud service). The work is in the context of a Qubit platform (a gate-model quantum computer). Taking into account the present limitation of hardware access, different quantum machine learning methods are studied on simulators and the results are compared with classical machine learning methods (BDT, classical Support Vector Machine and classical Neural Network). Furthermore, we do apply quantum machine learning on IBM quantum hardware to compare performance between quantum simulator and quantum hardware.

The work is performed by an international and interdisciplinary collaboration with the Department of Physics and Department of Computer Sciences of University of Wisconsin, CERN Quantum Technology Initiative, IBM Research Zurich, IBM T.J. Watson Research Center, Fermilab Quantum Institute, BNL Computational Science Initiative, State University of New York at Stony Brook, and Quantum Computing and AI Research of Amazon Web Services. This work pioneers a close collaboration of academic institutions with industrial corporations in the High Energy Physics analyses effort.

Though the size of event samples in future HL-LHC physics and the limited number of qubits pose some challenges to the Quantum Machine learning studies for High Energy Physics, more advanced quantum computers with larger number of qubits, reduced noise and improved running time (as envisioned by IBM and Google) may outperform classical machine learning in both classification power and in speed.

Although the era of efficient quantum computing may still be years away, we have made promising progress and obtained preliminary results in applying quantum machine learning to High Energy Physics.

The 2021 European Physical Society conference for high energy physics
July 26-30, 2021
Online

[*]Speaker.

[†]Partially supported by DOE DE-SC0020416

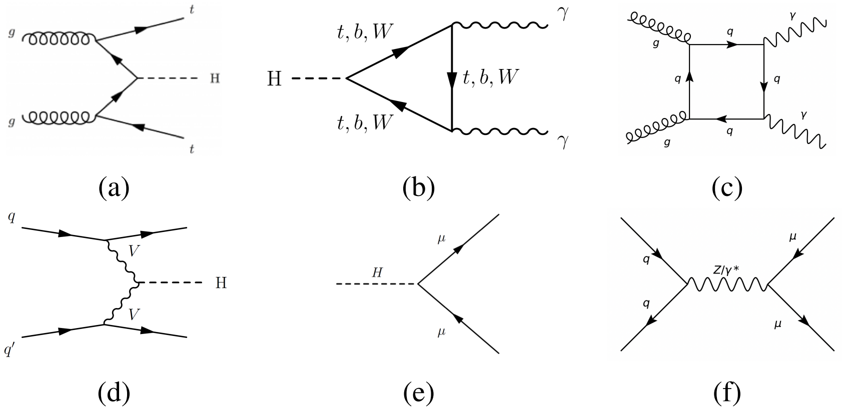
1. Introduction

The experimental programs of the LHC revolve around the discovery of new physics, which requires the use of machine learning algorithms for the identification of rare signals against immense backgrounds. The ambitious High Luminosity upgrade of the LHC (HL-LHC) will require an enormous amount of computing resources within the next two decades and beyond, and new technologies are being sought after to replace the present computing infrastructure. We aim to demonstrate as a proof of principle that quantum computing can be the new paradigm for HEP data analysis. Some authors of this paper belong to one of the LHC groups which have pioneered the use of classical machine learning in prominent physics analysis, two of which are $t\bar{t}H\,(H \to \gamma\gamma)$ and $H \to \mu^+\mu^-$. Three quantum machine learning methods, the variational quantum classifier, the quantum support vector machine kernel, and the quantum neural network, are investigated and compared to classical methods when applied to these two analyses.

2. Two recent LHC flagship physics analyses

The observation of $t\bar{t}H$ production (Higgs boson production in association with a top quark pair) in 2018 by the ATLAS and CMS experiments [1, 2] was a significant milestone for the understanding of fundamental particles and interactions, as it confirmed the interactions between the Higgs boson and the top quark, the heaviest known fundamental particle. As $t\bar{t}H$ accounts for only about 1% of the total Higgs boson production at the LHC, its observation was extremely challenging. Here we address a channel where the Higgs boson decays into two photons ($H \to \gamma\gamma$) and the two top quarks decay into jets. To ensure the results are as realistic as possible, we closely follow an analysis strategy similar to that employed by ATLAS [1]. Starting from reconstructed events with two photons and at least three jets, we train classifiers to separate the $t\bar{t}H\,(H \to \gamma\gamma)$ signal from the dominant background of this analysis, non-resonant two-photon production. The training uses 23 kinematic variables similar to those in [1]: the transverse momentum p_T, pseudo-rapidity η and b-tagging status of up to 6 leading jets, the magnitude of the missing transverse momentum, as well as the $p_T/m_{\gamma\gamma}$ ($m_{\gamma\gamma}$ denotes the invariant mass of the photon pair) and η of the two photons.

The search for $H \to \mu^+\mu^-$ decay (Higgs boson decay into two muons) has become one of the most important topics in the LHC physics program [3, 4], to probe the Higgs coupling to second-generation fermions. The main challenge results from the small $H \to \mu^+\mu^-$ decay branching ratio of about 0.02%, but with more future data, the LHC experiments could establish or exclude the Higgs-muon coupling, either of which would be an exciting discovery. Following an analysis strategy similar to that used in ATLAS, we divide reconstructed two-muon events into several n_j (jet multiplicity) channels, and focus on the $n_j \geq 2$ channel to target vector boson fusion (VBF) Higgs production, whose signature is two forward jets. We train classifiers to distinguish between the $H \to \mu^+\mu^-$ signal and the dominant background, the production of a pair of muons through the exchange of a Z boson or a virtual photon ($Z/\gamma^* \to \mu^+\mu^-$). The training is based on 13 kinematic variables similar to those in [3]: the p_T and rapidity Y of the two-muon system, the absolute value of the cosine of the lepton decay angle in the Collins-Soper frame, the p_T and η of the two leading jets, the relative azimuthal angle of each jet with respect to the di-muon system, the p_T, Y and invariant mass of the two-jet system, and the relative azimuthal angle between the two-jet system



(a) (b) (c)

(d) (e) (f)

Figure 1: Representative Feynman diagrams for (a) $t\bar{t}H$ production, (b) $H \to \gamma\gamma$ decay, (c) non-resonant two-photon production, (d) VBF Higgs production, (e) $H \to \mu^+\mu^-$ decay, and (f) $Z/\gamma^* \to \mu\mu$ production. In these diagrams, H denotes a Higgs boson, g a gluon, q a quark, t a top quark, b a bottom quark, μ a muon, W a W boson, Z a Z boson, V a W boson or Z boson, and γ a photon.

and the two-muon system. Figure 1 shows representative Feynman diagrams for the signals and backgrounds in the $t\bar{t}H$ and $H \to \mu^+\mu^-$ analyses. In both cases, we generate all events using Madgraph5_aMC@NLO [5] plus Pythia6 [6], with the center-of-mass energy of the proton-proton collisions set to 13 TeV. The ATLAS detector response is simulated with Delphes [7]. The events are then passed to the machine learning algorithms, whether classical or quantum.

3. Application of Three Quantum Machine Learning Algorithms to LHC Physics

3.1 Method 1: Variational Quantum Classifier

The Quantum variational classifier method, proposed in Ref. [8], includes data encoding via feature map circuits, a quantum variational circuit with parameters, a measurement and a conversion from the measurement to a classification label.

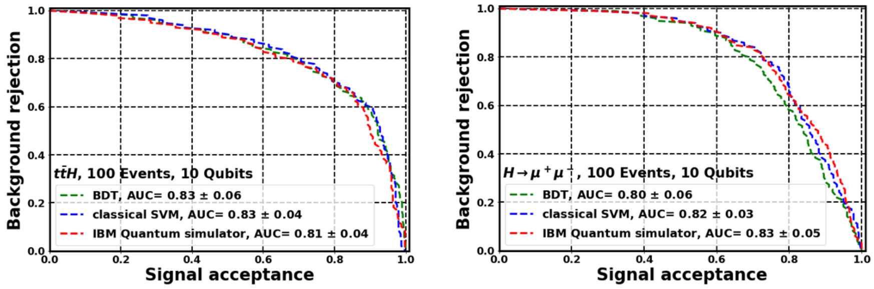
We studied this method using simulation and IBM quantum computer hardware with 10 qubits in Qiskit [9]. Ten independent datasets are trained and the performance are averaged. A classical SVM [10] and a classical BDT [11] are trained using the same dataset as benchmarks for classical machine learning algorithms. Figure 2 shows comparable receiver operating characteristic (ROC) curves among the three methods, for both the $t\bar{t}H$ and $H \to \mu^+\mu^-$ analyses. The values of the area under the ROC curve (AUC) are also shown. Figure 3 shows the results comparing the quantum simulation and hardware.

3.2 Method 2: Quantum Support Vector Machine Kernel Method

The quantum support vector machine kernel method, or QSVM-Kernel, introduced in Refs. [8, 12], leverages the quantum state space as a feature space to efficiently compute kernel entries. We employed QSVM-Kernel algorithm on the $t\bar{t}H$ analysis, using up to 20 qubits on the quantum computer simulators and up to 15 qubits on the quantum computer hardware.

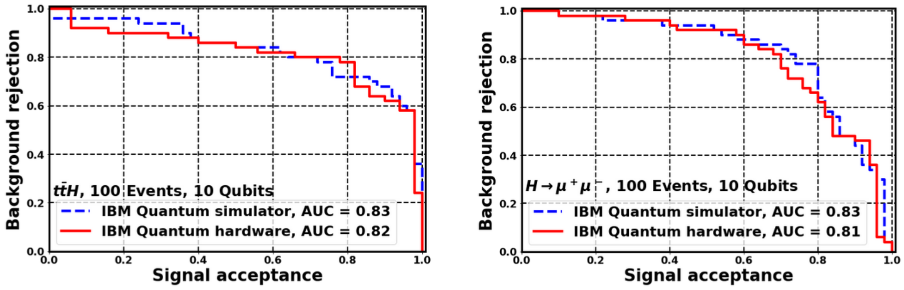
PoS(EPS-HEP2021)842

PoS(EPS-HEP2021)842



(a) (b)

Figure 2: ROC curves of the quantum variational classifier method on the ibmq QasmSimulator (blue), the classical SVM (yellow), and the BDT (green) for (a) the ttH and (b) the $H \rightarrow \mu^+\mu^-$ analyses.



(a) (b)

Figure 3: ROC curves of the quantum variational classifier method showing comparison between the quantum simulator and quantum hardware for (a) the $t\bar{t}H$ and (b) the $H \rightarrow \mu^+\mu^-$ analyses.

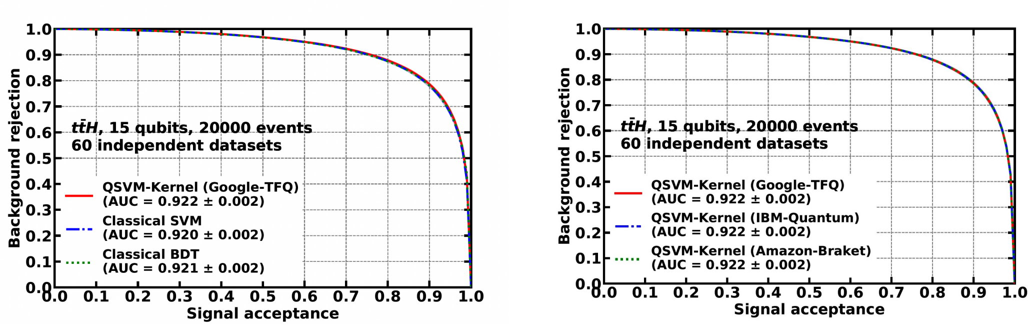
The algorithm maps the input feature vector, $\vec{x}$, non-linearly to a quantum state of N qubits, $\phi(\vec{x})$, by applying a quantum feature map circuit. The similarity of each input event pair is computed as the the inner product of the quantum states, $K(\vec{x}_1,\vec{x}_2) = |\langle\phi(\vec{x}_1)|\phi(\vec{x}_2)\rangle|^2$. The kernel entries are then used to find an optimal separating hyperplane that separates signal from background. The QSVM-Kernel algorithm is implemented in Google TensorFlow-Quantum [13], IBM Qiskit [9] and Amazon Braket [14]. The performance with increasing the number of qubits and the number of training events are studied. For each configuration, 60 independent datasets are trained and the performance are averaged to reduce the statistical fluctuations.

The QSVM-Kernel algorithm gives comparable performance to its classical counterpart, SVM, as well as the classical BDT, as shown in Figure 4(a). Figure 4(b) overlays the results from TensorFlow-Quantum, Qiskit and Bracket frameworks using 15 qubits and 20000 training events, all of which gives close performance.

In Figure 5, the ROC curve using the “ibmq paris” hardware is compared to the simulation using 15 qubits, 100 events. The performance achieved by the quantum computer hardware is promising and approaching the noiseless quantum computer simulator.

3.3 Method 3: Quantum Neural Network

The quantum neural networks (QNNs) bridges the neural network algorithms with the quan-



(a) (b)

Figure 4: ROC curves of various classifiers in the $t\bar{t}H$ analysis using 15 qubits and 20000 events. Each curve represents results averaged over 60 statistically independent datasets.

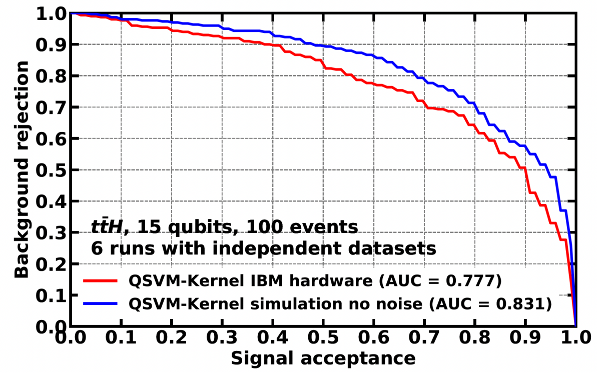


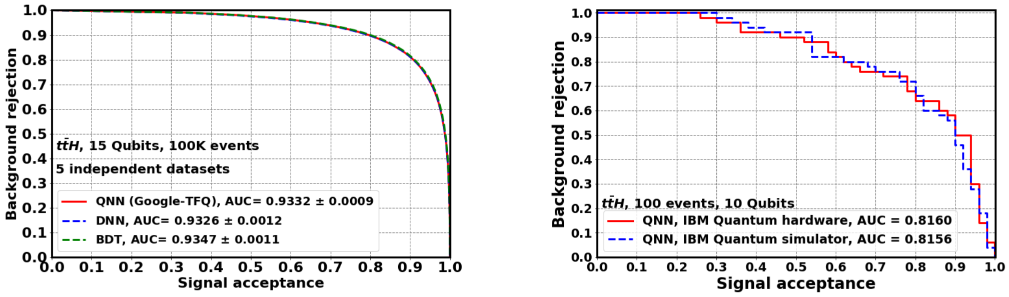
Figure 5: ROC curves of the QSVM-Kernel classifier with the “ibmq paris” quantum computer hardware and the noiseless quantum computer simulator using the $t\bar{t}H$ analysis datasets of 15 qubits and 100 events.

tum computing. It is designed to replace the computational intensive part of a classical NN algorithm to by a quantum algorithm that can be performed on a quantum computer, with the aim of a better efficiency and performance. The large latent space represented by the qubits in QNNs may potentially lead to a better global minimum than the classical neurons in classical NNs. This field attracted enormous attentions from researchers, with many works done using the existing software libraries such as TensorFlow Quantum [13], Qiskit [9], etc.

To fit to the NISQ devices, a hybrid QNN of three layers is explored in this study. The first layer is a classical dense layer to transform the input features with higher dimensions to lower dimensions such that they can be encoded in a low number of available qubits. The second layer is the quantum layer–the core part of the QNN architecture. The quantum layer consists of a featuremap circuit encoding the classical input into a quantum state, a variational circuit containing the trainable parameters, and the measurement measuring the quantum states. The output from the first layer is encoded into the qubits with the featuremap circuit and then a variational circuit with trainable parameters (parametrised circuits) is applied. Measurement is performed on all the qubits. Results are then plugged to a second classical dense layer to convert to a classification label. The

PoS(EPS-HEP2021)842

classical neuron weights and the parameters associated with the quantum circuits are updated using a gradient-based method during the training.



(a) (b)

Figure 6: ROC curves of various classifiers using the $t\bar{t}H$ analysis datasets of 100000 events and 23 input variables (a), ROC curves of the QNN classifier with the "ibmq paris" quantum computer hardware and IBM quantum computer simulator using the $t\bar{t}H$ analysis datasets of 100 events (b).

This computing model is constructed using TensorFlow Quantum library [13] and applied to the $t\bar{t}H$ analysis. The input events are divided into five datasets, each with a training sample, a validation sample, and a test sample. With each dataset, a 15 qubits QNN model, a classical Deep Neural Network (DNN) model, and a classical Boosted Decision Tree (BDT) mode are trained and evaluated. The final performance is an average over the five datasets. Comparable results are observed among the three models, as is shown in Figure 6(a).

To demonstrate the power of the QNN algorithm, a training with 100 events and 10 qubits is performed on one of the IBM quantum computers. The performance from the quantum computer hardware (ibmq paris) is similar to the simulation using the same number of events and qubits as is shown in Figure 6(b).

4. Summary

In summary, we have successfully employed three types of quantum machine learning algorithms: the variational quantum classifier, the quantum support vector machine kernel method, and the quantum neural network, focusing on the $t\bar{t}H$ $(H \to \gamma\gamma)$ and $H \to \mu^+\mu^-$ LHC HEP flagship analyses with Delphes simulation events. Our results on both quantum simulators and hardware demonstrate that quantum machine learning on gate-based quantum computers, where computing is achieved by a sequence of quantum gates as opposed to D-wave quantum annealers, already has the ability to differentiate between signal and background in realistic physics datasets.

The greatest present challenge of using quantum computer hardware for machine learning in future HL-LHC physics is the size of the event samples and the number of qubits. Currently, each hardware job uses way less events than simulation due to the limitation of the access time. The number of qubits to use is restricted, as well as the circuit length and number of entanglement gates in the algorithms. As far as the authors are aware at the time of writing, the maximum numbers of qubits available from IBM or Google quantum hardware are 65 and 54, respectively.

PoS(EPS-HEP2021)842

In the context of HEP data analysis, future studies will aim to show that quantum machine learning outperforms classical machine learning in classification power. In simulations, we are confident that this may be achieved by exploiting a larger number (e.g., ≥ 30) of qubits. Furthermore, more advanced quantum computers may offer increases in speed for quantum machine learning, which is critical for such HEP applications. Indeed, roadmaps presented by IBM and Google suggest that future quantum hardware will have reduced noise, improved running time, and achieve a performance close to noiseless quantum simulators, an expectation which is justified by the recent large investments and fierce international competition in quantum technology.

5. Acknowledgement

We would like to acknowledge and thank our collaborators in IBM Research Zurich and IBM T.J. Watson Research Center - P. Barkoutsos, I. Tavernelli, S. Woerner, and J. Glick - for their contribution to this proceeding. In particular, we are indebted to their extensive knowledge of the IBM Quantum simulator and IBM Quantum hardware.

References

[1] ATLAS Collaboration, *Observation of Higgs boson production in association with a top quark pair at the LHC with the ATLAS detector*, *Phys. Lett. B* 784, 173 (2018).

[2] CMS Collaboration, *Observation of ttH Production*, *Phys. Rev. Lett.* 120, 231801 (2018).

[3] ATLAS Collaboration, A Search for the Dimuon Decay of the Standard Model Higgs Boson with the ATLAS Detector. *Phys. Lett. B* 812, 135980 (2021).

[4] CMS Collaboration, Evidence for Higgs boson decay to a pair of muons. *JHEP* 01, 148 (2021).

[5] Alwall, J., Frederix, R., Frixione, S., Hirschi, V., Maltoni, F., Mattelaer, O., Shao, H. -S., Stelzer, T., Torrielli, P. and Zaro, M. The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations. *JHEP* 07, 079 (2014).

[6] Sjostrand, T., Mrenna, S. and Skands, P. Z. PYTHIA 6.4 physics and manual. *JHEP* 05, 026 (2006).

[7] de Favereau, J., Delaere, C., Demin, P., Giammanco, A., Lemaître, V., Mertens, A. and Selvaggi, M. DELPHES 3, a modular framework for fast simulation of a generic collider experiment. *JHEP* 02, 057 (2014).

[8] Havlíček, V., Córcoles, A. D., Temme, K., Harrow, A. W., Kandala, A., Chow, J. M. and Gambetta, J. M., *Supervised learning with quantum-enhanced feature spaces*, *Nature* 567, 209 (2019).

[9] Aleksandrowicz, G. et al., *Qiskit: An Open-source Framework for Quantum Computing. https://qiskit.org (2019).*

[10] Boser, B. E., Guyon, I. M., Vapnik, V. N., *A training algorithm for optimal margin classifiers, Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory, 144-152 (1992).*

[11] J. H. Friedman, *Stochastic gradient boosting, Computational Statistics Data Analysis*, *38, 367 (2002).*

[12] M. Schuld and N. Killoran, *Quantum machine learning in feature Hilbert spaces*, *Physical Review Letters 122, 040504 (2019).*

[13] M. Broughton et al., *TensorFlow Quantum: A Software Framework for Quantum Machine Learning*, *arXiv:2003.02989 (2020).*

[14] Amazon Braket, *https://aws.amazon.com/braket/ (2019).*

PoS(EPS-HEP2021)842