Mechanics of Advanced Composite Structures

Journal homepage: https://macs.semnan.ac.ir/

Semnan University

ISSN: 2423-7043

Research Article

Quantum Machine Learning Approach for the Prediction of Surface Roughness in Additive Manufactured Specimens

Akshansh Mishra [a] [0000-0003-4939-359X], **Vijaykumar S Jatti** [b*] [0000-0001-7949-2551]

[a] *School of Industrial and Information Engineering, Politecnico Di Milano, Milan, Italy*

[b*] *Symbiosis Institute of Technology, Symbiosis International (Deemed) University, Pune, Maharashtra, India*

ARTICLE INFO

Article history:

Received: 2023-03-22
Revised: 2023-05-12
Accepted: 2023-10-19

Keywords:

Additive manufacturing;
Quantum neural network;
Quantum forest;
Variational quantum classifier;
Fused deposition modeling.

ABSTRACT

One of the most important factors affecting the functioning and performance of additively produced components is surface roughness. Precise estimation of surface roughness is essential for streamlining production procedures and guaranteeing product quality. Recently, quantum computing has drawn interest as a possible way to solve challenging issues and produce accurate prediction models. For the first time, we compare three quantum algorithms in-depth in this research paper for surface roughness prediction in additively manufactured specimens: the Quantum Neural Network (QNN), Quantum Forest (Q-Forest), and Variational Quantum Classifier (VQC) modified for regression. Mean Squared Error (MSE), Mean Absolute Error (MAE), and Explained Variance Score (EVS) are the assessment metrics we use to evaluate the algorithms' performance. With an MSE of 56.905, an MAE of 7.479, and an EVS of 0.2957, the Q-Forest algorithm outperforms the other algorithms, according to our data. On the other hand, the QNN method shows a negative EVS of -0.444 along with a higher MSE of 60.840 and MAE of 7.671, suggesting that it might not be the best choice for surface roughness prediction in this application. The regression-adapted VQC has an MSE of 59.121, an MAE of 7.597, and an EVS of -0.0106, indicating that it performs inferior to the Q-Forest approach as well.

1. Introduction

In recent years, there has been a significant interest in quantum computing, a new discipline that uses the concepts of quantum mechanics to transform a variety of sectors. [1] Quantum computing uses qubits, which can coexist in numerous states simultaneously because of the phenomena of superposition and entanglement, in contrast to classical computing, which uses bits to encode information as either 0 or 1 [2]. Because of this special property, quantum computers are able to execute complex computations tenfold quicker than traditional computers, opening up new avenues for problem-solving [3].

The manufacturing industry, characterized by its elaborate and complex processes, stands to benefit significantly from advancements in quantum computing[4-5]. Potential applications of quantum computing in manufacturing cover a wide range of areas, including optimization, simulation, material science, and predictive modeling. Integrating quantum computing into these fields could lead to considerable improvements in efficiency, cost reduction, and product quality [6].

Quantum computing can optimize various manufacturing processes, such as scheduling,

* **Corresponding author.**
E-mail address: *vijaykumar.jatti@sitpune.edu.in*

Cite this article as:
Mohammadi, A., Mahdi-Nia, M., 2024. Title of article. *Mechanics of Advanced Composite Structures*, 11(2), pp. 1402-1425
https://doi.org/10.22075/MACS.2023.39315.2050

resource allocation, and supply chain management, by identifying the most efficient and cost-effective solutions [7]. In complex, large-scale scenarios, quantum optimization algorithms like Quantum Approximate Optimization Algorithm (QAOA) and Quantum Adiabatic Computing may outperform classical optimization techniques [8]. The enhanced capacity of quantum computers to simulate quantum systems has allowed researchers to get deeper insight into materials and compounds at the atomic and molecular levels [9]. With this increased understanding, innovative materials and cutting-edge manufacturing techniques may be developed faster, resulting in more efficient and sustainable production processes [10].

In the domain of material science, quantum computing can expedite the discovery and design of novel materials with customized properties by simulating their behavior at the quantum level [11]. This could lead to the creation of advanced materials with improved strength, durability, and energy efficiency, thereby enhancing the overall quality of manufactured products [12]. Quantum computing can also significantly improve predictive modeling in manufacturing by providing more accurate and efficient solutions for intricate problems. Critical metrics like machine wear, defect rates, and surface roughness that impact product quality and process efficiency can be predicted using quantum machine learning algorithms. The Quantum Neural Network (QNN), Quantum Forest (Q-Forest), and Variational Quantum Classifier (VQC) are a few examples of these algorithms. These algorithms can decrease production errors, decrease downtime, and enhance maintenance schedules. Quantum machine learning algorithms like QNN, Q-Forest, and VQC offer promising approaches to predict surface roughness in additively manufactured parts. Their ability to handle complex, high-dimensional data makes them suitable for this task, potentially providing more accurate and efficient predictions than classical methods [13].

In the past researchers have predicted the surface roughness of AM parts using linear regression [14], multiple linear regression [15], polynomial regression [16], response surface methodology [17], and the Taguchi method [18].

In additive manufacturing, surface roughness is essential to a part's mechanical, functional, and aesthetic properties. Surface roughness can be optimized and controlled to increase manufacturing process efficiency, lower costs, and improve product performance. As a result, it is crucial to take into account during the additive manufacturing design and production phases. Focusing on AM components not only addresses specific industry needs but also pushes the boundaries of predictive modeling, paving the way for more efficient and reliable manufacturing practices.

Quantum Machine Learning (QML) algorithms like Quantum Neural Networks (QNN), Quantum Forests (Q-Forest), and Variational Quantum Classifiers (VQC) offer novel and promising approaches to predict surface roughness in additively manufactured parts. These quantum algorithms leverage the principles of quantum mechanics to process and analyze data more efficiently and accurately than classical machine learning methods, providing significant advantages in the field of additive manufacturing. The ability of QNNs to efficiently model and predict surface roughness with high accuracy, due to their inherent parallelism and capacity to handle high-dimensional data, represents a significant advancement over traditional neural networks. The introduction of quantum decision trees and their ensemble approach in Q-Forest offers a novel method for handling the variability and complexity of additive manufacturing data, leading to improved prediction accuracy for surface roughness. VQCs introduce a novel classification mechanism that uses the quantum state space for enhanced pattern recognition and classification accuracy in predicting surface roughness, surpassing the capabilities of classical classifiers.

2. Quantum Machine Learning Framework

The goal of the developing multidisciplinary discipline of quantum machine learning is to combine machine learning and quantum computing to take advantage of the special qualities of quantum systems for solving optimization problems and complex calculations that are more effective than those of classical computing. As this field is still in its nascent stage, its potential applications are vast and varied. To comprehend the intricate workings of quantum machine learning, it is essential to understand several fundamental concepts of Quantum bits (qubits), Quantum entanglement, and Quantum gates.

Unlike classical bits, which can only exist in one of two states, qubits in quantum computing are the fundamental units that can exist in a superposition of both 0 and 1 states. In Fig. 1, the basis states |0> and |1> are represented as complex linear combinations to represent qubits. This characteristic makes parallelism possible and permits quantum computers to handle large amounts of data at once.

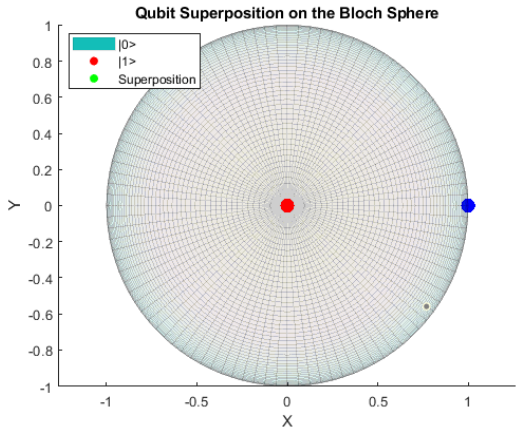

Fig.1. Schematic representation of Quantum Superposition [1]

To explain this with equations, let's consider a qubit, represented as $|\psi\rangle$ as depicted in Equation 1.

$$|\psi\rangle=\alpha|0\rangle+\beta|1\rangle \quad (1)$$

Here, the probability amplitudes of the qubit being in states $|0\rangle$ or $|1\rangle$, respectively, are determined by the complex coefficients α and β. Equations 2 and 3 provide the probabilities that the qubit is in states $|0\rangle$ or $|1\rangle$.

$$P(|0\rangle)=|\alpha|2 \quad (2)$$
$$P(|1\rangle)=|\beta|2 \quad (3)$$

Since the qubit must be in either state $|0\rangle$ or $|1\rangle$, the sum of the probabilities must be equal to 1 as depicted in Equation 4.

$$|\alpha|2+|\beta|2=1 \quad (4)$$

This property of qubits allows them to exist in a superposition of states and enables quantum parallelism. In a quantum computer with n qubits, there can be 2n different states that can be processed simultaneously. This is because the state of an n-qubit system can be written as Equation 5.

$$|\psi n\rangle=\Sigma ici|i\rangle \quad (5)$$

where i ranges from 0 to (2n - 1) and $\Sigma i\ |ci|2 = 1$.

This parallelism allows quantum computers to perform complex computations that are infeasible for classical computers.

Quantum entanglement is an exceptional phenomenon that occurs when two or more qubits become intricately interconnected, rendering them inseparable, as shown in Fig. 2. This can be represented by the entangled state Equation 6.

$$|\Psi\rangle=\alpha|00\rangle+\beta|11\rangle \quad (6)$$

where $|\Psi\rangle$ is the entangled state, α and β are complex coefficients, and $|00\rangle$ and $|11\rangle$ are the basis states of the two entangled qubits. In this scenario, the state of one qubit cannot be described independently of the other, as the probabilities of the outcomes are determined by the combined state $|\Psi\rangle$.

Entanglement is a crucial resource in quantum computing, as it enables non-local correlations and augments computational capabilities. One example of this is the Bell states, which are a set of maximally entangled states that can be written as Equation 7.

$$|\Phi+\rangle = (1/\sqrt{2})(|00\rangle + |11\rangle)$$
$$|\Phi-\rangle = (1/\sqrt{2})(|00\rangle - |11\rangle)$$
$$|\Psi+\rangle = (1/\sqrt{2})(|01\rangle + |10\rangle)$$
$$|\Psi-\rangle=(1/\sqrt{2})(|01\rangle-|10\rangle) \quad (7)$$

These Bell states demonstrate the strong correlations that can be achieved through quantum entanglement, which can be leveraged in various quantum algorithms, communication protocols, and teleportation, thus significantly enhancing the computational power of quantum systems.

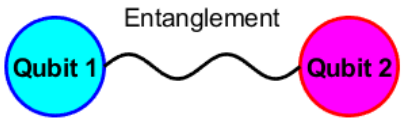

Fig. 2. The entanglement between the qubits

As the foundation of quantum circuits, quantum gates perform operations on qubits. Analogous to classical logic gates (e.g., AND, OR, NOT), quantum gates are reversible and can act on quantum state superpositions shown in Fig. 3.

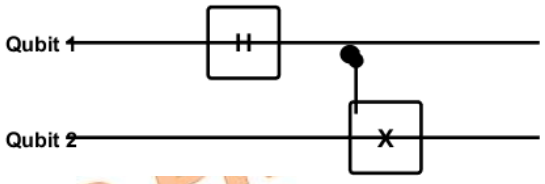

Fig. 3. Quantum circuit with two qubits, a Hadamard gate (H) acting on Qubit 1, and a CNOT gate acting on both qubits. [2]

The detailed mechanism of quantum machine learning is shown in Fig. 4.

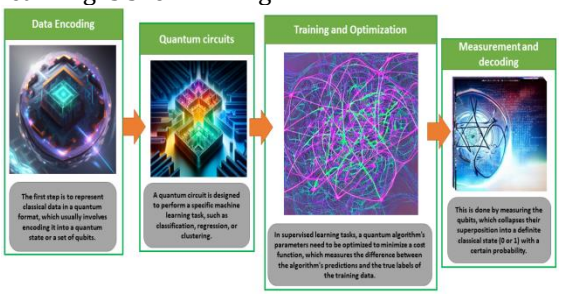

Fig. 4. Framework of Quantum Machine Learning

The initial step involves converting classical data into a quantum format, typically by encoding it into a quantum state or a set of qubits. This process employs quantum feature maps or embedding techniques that transform classical data into a high-dimensional Hilbert space, making it compatible with quantum computing. A quantum circuit tailored to a specific machine learning task, such as classification, regression, or clustering, is designed by selecting suitable quantum gates and arranging them sequentially to process input data and produce the desired output. The circuit's structure and complexity may vary according to the problem and required accuracy level. For supervised learning tasks, quantum algorithm parameters must be optimized to minimize a cost function that measures the discrepancy between algorithm predictions and the true labels of the training data. Quantum algorithms often employ variational techniques in which a classical optimization algorithm updates the quantum circuit parameters based on feedback from the quantum computer. Upon processing input data through the quantum circuit, results must be extracted by measuring qubits, causing their superposition to collapse into a definite classical state (0 or 1) with specific probabilities. The classical output is then decoded and interpreted to provide insights or predictions concerning the original problem.

3. Materials and Methods

The ASTM E8 standard geometry underwent a uniform 50% reduction in dimensions to maintain consistency throughout the model. The size of the sample is 87.50 mm x 9.884 mm x 5 mm. This reduction resulted in a drop in print size, material consumption, and time. The Response Surface Methodology (RSM) Design of the Experiment was used to establish thirty distinct trial conditions, each with three input parameter values (see Fig. 5). Using Ultimaker Cura software, the CAD model was sliced to generate G-code (see Fig. 6). For the experimental investigation, the Creality 3D FDM printer (see Fig. 7) was utilized. The material utilized was polylactic acid (PLA), and different parameters were applied to each print, such as layer height (0.1-0.2 mm), wall thickness (1-3 mm), print speed (40-120 mm/s) infill density (50%-100%), infill pattern (Honey comb-Triangular), bed temperature (60-80 ^{0}C), and nozzle temperature (200-220 ^{0}C). A digital vernier caliper was used to measure the length variations between each model and the original CAD file, and an input parameter datasheet was generated.

After being transformed into a CSV file, the experiment's data was imported into the Google Colab platform. Three distinct quantum machine learning algorithms—Quantum Neural Network (QNN), Quantum Forest (Q-Forest), and Variational Quantum Classifier (VQC)—that have all been modified for regression analysis will be implemented on this platform. Three metrics—Mean Square Error (MSE), Mean Absolute Error (MAE), and Explained Variance Score—were used to assess the effectiveness of these algorithms.

Fig. 5. Fabricated Additive Manufactured specimens [19]

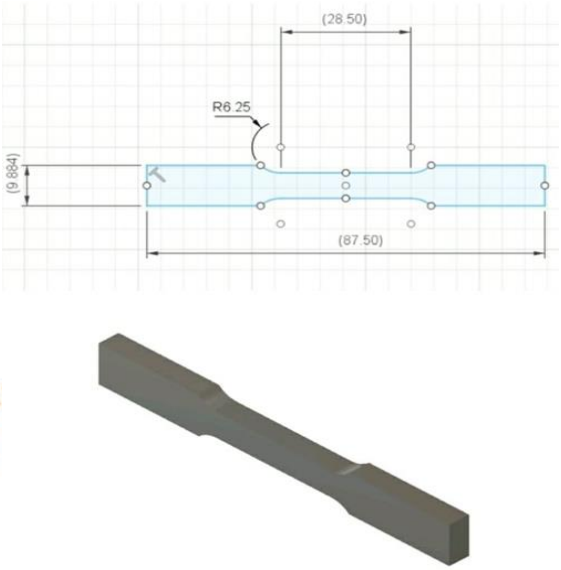

Fig. 6. Schematic sketch of the specimen [19]

4. Results and Discussion

The results of the Surface roughness measurements obtained through various input parameter combinations are presented in Table 1.

Table 1. Experimental results [20-23]

Layer height (in mm)	Wall thickness (in mm)	Infill density (in %)	Infill pattern	Nozzle temperature (in 0C)	Bed temperature (in 0C)	Print speed (in mm/s)	Fan speed (in %)	Surface Roughness, Ra (in µm)
.1	= 1	50	Honeycomb	200	60	120		6.13
....	....	....	..	...	...	....	....	...
....	....	....	..	...	...	....	....	...
....	....	....	..	...	...	....	....	...
0.2	3	60	Triangles	220	80	40	100	9.382

4.1 Quantum Neural Network (QNN)

Quantum Neural Networks (QNNs) represent an innovative approach that merges the principles of quantum computing with classical neural network architectures to address complex problems more effectively. Utilizing quantum bits (qubits) as the primary information carriers, QNNs leverage the inherent quantum properties of superposition to allow for simultaneous representation of multiple states.

At the core of QNNs lie quantum layers, which consist of sequential quantum gates. These gates are the fundamental mathematical components of quantum computing, responsible for transforming qubit states. Quantum gates are represented by unitary matrices, ensuring that the normalization of qubit states is maintained. Consequently, a quantum circuit is formed through a series of quantum gates, with the overall transformation resulting from the product of the matrices that represent the individual gates.

In mathematical terms, a quantum state can be expressed as a complex vector in a Hilbert space, where each element corresponds to the probability amplitude of a distinct computational basis state. The quantum state comprising n qubits can be denoted as Equation 8.

$$|\psi\rangle=\Sigma(ai|i\rangle) \quad (8)$$

Here, i ranges from 0 to 2n - 1, ai signifies complex coefficients, and |i⟩ represents the corresponding computational basis state. The probability of measuring the state |i⟩ is given by the square of the modulus of ai.

The transformation of a quantum state through a quantum gate can be depicted as a matrix-vector multiplication as depicted in Equation 9.

$$|\psi'\rangle=U|\psi\rangle \quad (9)$$

In the given equation, U stands for the unitary matrix associated with the quantum gate, |ψ> for the initial state, and |φ'> for the final state following the application of the gate.

A quantum circuit is run on a simulator or a quantum computer to ascertain a QNN's output. As seen in Fig. 7, the quantum circuit is operated on a simulator in our current study. A classical neural network or other classical machine learning models process the resulting output, which is a probability distribution across the computational basis states, to produce the final result.

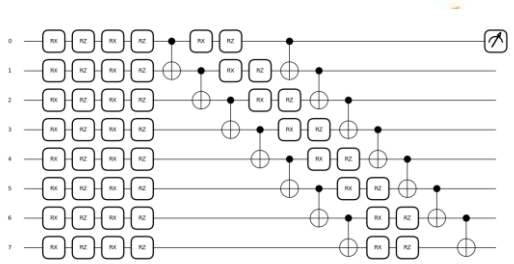

Fig. 7. Quantum Circuit framework in the present work [2]

The QNN is defined by the function qnn(params, x=None), which takes two arguments: params representing the network's parameters, and an optional input feature vector x. Within the QNN, the input features are embedded into the quantum state using a series of rotations (qml. RX and qml. RZ). These rotations act on the qubits of the quantum circuit, effectively encoding the classical data into the quantum system.

The quantum layers are defined in a loop that iterates through the params array. For each layer, rotations are applied to the qubits using the parameters in the array, followed by a series of controlled-NOT (CNOT) gates that create entanglement between adjacent qubits. This entanglement allows for the exploration of a larger solution space and enhances the network's expressivity. The output of the QNN is obtained by measuring the expectation value of the Pauli-Z operator on the first qubit. This expectation value represents the network's prediction for the given input.

The cost function, cost(params, X, y), calculates the MSE between the QNN's predictions and the target values. It does this by iterating through the input data X and calling the QNN function with the current parameters and input features. The mean squared error between the predictions and the ground truth targets y is then computed and returned as the cost value.

The QNN's parameters are initialized randomly and optimized using the gradient descent optimizer provided by the Pennylane library. The optimization is performed for 100 iterations, with the parameters updated at each step to minimize the cost function. The test set is predicted using the trained QNN, and its performance is assessed using the three metrics indicated in Table 2: mean squared error (MSE), mean absolute error (MAE), and explained variance score.

Table 2. Obtained Metrics features for QNN Algorithm

MSE	MAE	Explained Variance Score
60.840	7.671	-0.444

The plot of explained variance score vs training iterations is typically used to track the performance of a machine learning algorithm during the training process as shown in Fig. 8. A metric called the explained variance score quantifies the percentage of the target variable's volatility that the model can account for. It has a range of 0 to 1, where 1 means that all of the variance is explained by the model and 0 means that none of it is. The plot of explained variance score vs training iterations can provide insights into how well the model is learning and converging to a solution. If the explained variance score is increasing with each iteration, it suggests that the model is improving and learning from the data. On the other hand, if the explained variance score is plateauing or decreasing, it suggests that the model is not improving or may be overfitting the data.

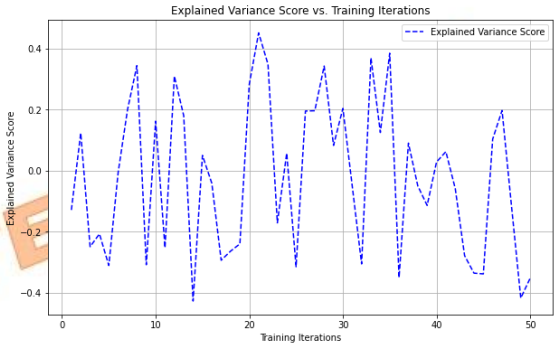

Fig. 8. Explained variance score vs training iterations of QNN algorithm.

4.2 Q-Forest Algorithm

The Q-Forest algorithm, a quantum-inspired approach, is employed for clustering and classification tasks in large-scale data processing. Drawing upon principles of quantum mechanics such as quantum entanglement and quantum superposition, the algorithm enhances both efficiency and effectiveness. It begins by converting classical data into quantum states, assuming a dataset comprising N samples and d features, with each sample represented as a d-dimensional vector. Subsequently, the algorithm transforms classical data points into quantum states, generating a quantum superposition of the dataset, which enables concurrent manipulation of all data points. The algorithm calculates pairwise distances between data points using a distance metric to establish a quantum entanglement representation of the data as shown in Fig. 9. These pairwise distances are encoded into quantum states, such that the entangled states depict the relationships among data points. This encoding method permits the algorithm to evaluate distances between data points more effectively than classical approaches.

The core procedure of the Q-Forest algorithm entails the construction of a quantum decision tree, in which data points are recursively divided into subsets until a termination criterion is satisfied. The quantum decision tree is formed by determining the optimal split point for each node to minimize an impurity measure. Quantum parallelism is employed during this process to efficiently explore all potential split points. Upon completion of the tree, it can be utilized to cluster or classify novel data points based on their quantum entanglement representation.

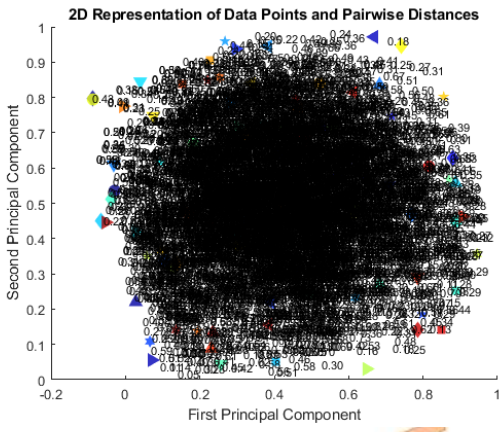

Fig. 9. Calculation of pairwise distances between data points using a distance metric to establish a quantum entanglement

The obtained metric features are represented in Table 3. Figure 10 shows the variation of the Explained variance score with the training iterations.

Table 3. Obtained Metrics features for Q-Forest Algorithm

MSE	MAE	Explained Variance Score
56.905	7.479	0.2957

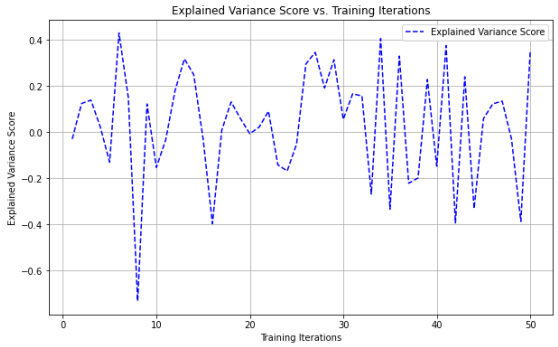

Fig.10. Explained Variance score vs training iterations for Q-Forest algorithm

4.3 Variational Quantum Classifier (VQC) adapted for regression

Regression tasks are carried out by a hybrid quantum-classical machine learning algorithm called a Variational Quantum Classifier (VQC), which leverages quantum circuits. This method prepares a parametrized quantum circuit with movable parameters denoted by θ. The quantum circuit functions as a feature map, encoding the input data into a high-dimensional quantum state. The actual regression is then performed on the encoded data by a second parametrized quantum circuit called the variational circuit. The expectation value of a particular observable is measured to provide a continuous value as the variational circuit's output. The loss function is modified to assess the discrepancy between the actual target values and the anticipated continuous values in order to adapt VQC for regression. In a traditional optimization loop, this loss function is employed to optimize the variational circuit's parameters. The VQC model learns the underlying pattern in the data and develops the ability to anticipate values for data points that have not yet been seen by iteratively changing the circuit parameters. The obtained metric features are represented in Table 4. Figure 11 shows the variation of the Explained variance score with the training iterations.

Table 4. Obtained Metrics features for Variational Quantum Classifier (VQC) adapted for regression

MSE	MAE	Explained Variance Score
59.121	7.597	-0.0106

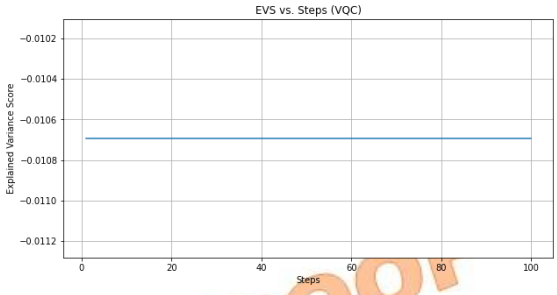

Fig. 11. Explained Variance score vs training iterations for Variational Quantum Classifier (VQC) adapted for regression

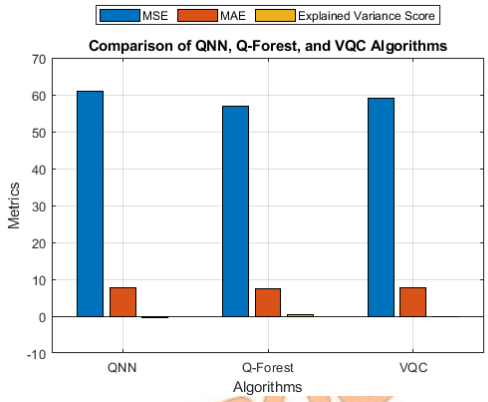

Fig.12. Comparison of the implemented algorithms

Figure 12 shows the overall comparison of the obtained results. Our results indicate that the Q-Forest algorithm outperforms the other two algorithms in terms of both MSE and MAE. Q-Forest achieved an MSE of 56.905 and an MAE of 7.479, while the QNN and VQC algorithms recorded an MSE of 60.840 and 59.121, and an MAE of 7.671 and 7.597, respectively. Lower MSE and MAE values indicate better performance in terms of prediction accuracy, demonstrating that Q-Forest is better suited for this particular regression task compared to the other algorithms. Also, the Explained Variance Score (EVS) shows that the Q-Forest algorithm accounts for 29.57% of the total variance in the data, whereas the QNN and VQC algorithms record negative EVS values of -44.4% and -1.06%, respectively. A higher EVS value suggests that the model can better explain the variance in the dataset, and therefore, the Q-Forest algorithm demonstrates superior performance in this aspect as well. These findings suggest that the Q-Forest algorithm is a more effective approach for solving regression tasks compared to the QNN and VQC algorithms, in the context of the dataset and problem studied. However, it is essential to note that the performance of quantum algorithms can vary depending on the specific problem, dataset, and hyperparameter settings. As such, further research is necessary to explore the generalizability of our findings to different datasets and regression tasks. Additionally, it would be interesting to examine

the performance of these quantum algorithms in comparison to classical machine learning algorithms, which could offer valuable insights into the advantages and limitations of quantum computing in the field of regression analysis.

5. Conclusions

This study's main goal was to evaluate the accuracy of three quantum algorithms—QNN, Q-Forest, and VQC—adapted for regression in terms of surface roughness prediction for additively built specimens. Our investigation showed that the Q-Forest method performed the best, with an EVS of 0.2957, an MAE of 7.479, and an MSE of 56.905. This suggests that, in comparison to the other two methods, the Q-Forest algorithm not only produces predictions that are more accurate but also explains a larger percentage of the dataset's variance. With a negative EVS of -0.444 a higher MSE of 60.840 and an MAE of 7.671, the QNN algorithm may not be the best choice for estimating the surface roughness of additively built specimens. Likewise, the regression-adapted VQC obtained an MSE of 59.121, an MAE of 7.597, and an EVS of -0.0106, demonstrating that its performance is below that of the Q-Forest technique as well. Q-Forest's ensemble learning approach, using multiple quantum decision trees, enhances prediction accuracy by aggregating diverse models' outputs, leading to robust and reliable results. The algorithm leverages quantum parallelism, enabling it to process and analyze large datasets more efficiently than classical methods.

Future Directions includes:

a) Algorithm Refinement: Further research to refine QNN and VQC architectures and training processes for better regression performance.
b) Hybrid Models: Exploration of hybrid quantum-classical models to leverage the strengths of both approaches in predicting surface roughness.
c) Domain-Specific Customization: Customizing quantum algorithms to better suit the specific requirements and characteristics of additive manufacturing datasets.

References

[1] Steane, A., 1998. Quantum computing. Reports on Progress in Physics, 61(2), p.117.

[2] National Academies of Sciences, Engineering, and Medicine, 2019. Quantum computing: progress and prospects.

[3] Rawat, B., Mehra, N., Bist, A.S., Yusup, M. and Sanjaya, Y.P.A., 2022. Quantum computing and ai: Impacts & possibilities. ADI Journal on Recent Innovation, 3(2), pp.202-207.

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

[5] Suzuki, Y., Endo, S., Fujii, K. and Tokunaga, Y., 2022. Quantum error mitigation as a universal error reduction technique: applications from the nisq to the fault-tolerant quantum computing eras. PRX Quantum, 3(1), p.010345.

[6] Guijo, D., Onofre, V., Del Bimbo, G., Mugel, S., Estepa, D., De Carlos, X., Adell, A., Lojo, A., Bilbao, J. and Orus, R., 2022. Quantum artificial vision for defect detection in manufacturing. arXiv preprint arXiv:2208.04988.

[7] Yonaga, K., Miyama, M., Ohzeki, M., Hirano, K., Kobayashi, H. and Kurokawa, T., 2022. Quantum optimization with Lagrangian decomposition for multiple-process scheduling in steel manufacturing. Isij International, 62(9), pp.1874-1880.

[8] Permin, E., Borgard, S., Castillo Velasquez, L. and Pyschny, N., 2022. A Simple Approach for Complexity Reduction in Job Shop Scheduling Using Quantum Computers. Available at SSRN 4259131.

[9] Zinner, M., Dahlhausen, F., Boehme, P., Ehlers, J., Bieske, L. and Fehring, L., 2022. Toward the institutionalization of quantum computing in pharmaceutical research. Drug Discovery Today, 27(2), pp.378-383.

[10] Nakano, A., 2023. Exascale Simulations of Quantum Materials Guided by AI and Quantum Computing. Bulletin of the American Physical Society.

[11] Sennane, W., Rancic, M., Greene-Diniz, G., Zsolt Manrique, D., Magnin, Y., Cordier, P., Llewellyn, P., Krompiec, M., Muñoz Ramo, D. and Shishenina, E., 2023. Modelling carbon capture on metal-organic frameworks with quantum computing. Bulletin of the American Physical Society.

[12] Orth, P., Mukherjee, A., Yao, Y.X., Huynh, A. and Trevisan, T., 2023. Quantum computing simulation of nonlinear optical response in Hubbard models. Bulletin of the American Physical Society.

[13] Matsuo, S. and Souma, S., 2023. A proposal of quantum computing algorithm to solve Poisson equation for nanoscale devices under Neumann boundary condition. Solid-State Electronics, 200, p.108547.

[14]Guo, N., and Leu, M. C., 2013. Additive manufacturing: technology, applications and research needs. Frontiers of Mechanical Engineering, 8(3), pp. 215-243.

[15] Mahesh, M., and Basavarajappa, S., 2015. Surface roughness prediction in turning of titanium alloy by machine vision system. Procedia Engineering, 97, pp.369-376.

[16] Kumar, P., Singh, R., and Ahuja, I. P. S., 2012. Investigations on surface roughness in fused deposition modeling (FDM) using predictive modeling techniques. Journal of Manufacturing Processes, 14(3), pp.393-402.

[17] Singh, S., and Prakash, C., 2020. Optimization of process parameters to minimize surface roughness in FDM: A review and analysis. Journal of Manufacturing Processes, 49, pp. 92-107.

[18] Panda, B. N., Sahoo, S., Sahu, P. K., and Mahapatra, S. S., 2010. Optimization of fused deposition modelling (FDM) process parameters using Taguchi method. International Journal of Manufacturing Technology and Management, 21(1-2), pp.100-112.

[19] Potnis, M.S., Singh, A., Jatti, V.S. et al. Part quality investigation in fused deposition modelling using machine learning classifiers. Int J Interact Des Manuf (2023). https://doi.org/10.1007/s12008-023-01493-4

[20] Jatti, V. S., et al, 2022, Mechanical Properties of 3D-printed Components Using Fused Deposition Modeling: Optimization Using the Desirability Approach and Machine Learning Regressor', Applied System Innovation, Volume 5, Issue 6. https://www.mdpi.com/2571-5577/5/6/112

[21] Mishra, A., Jatti, V. S., 2023, Neurosymbolic artificial intelligence (NSAI) based algorithm for predicting the impact strength of additive manufactured polylactic acid (PLA) specimens', Engineering Research Express, Volume 5, Number 3 https://iopscience.iop.org/article/10.1088/2631-8695/ace610

[22] Mishra, A., Jatti, V.S., 2023. Novel Coupled Genetic Algorithm–Machine Learning Approach for Predicting Surface Roughness in Fused Deposition Modeling of Polylactic Acid Specimens. Jounral of Materials Engineering and Performance. https://doi.org/10.1007/s11665-023-08379-2

[23] Mishra, A. Jatti, V. S., Sefene, E.M., Paliwal, S., 2023. Explainable Artificial Intelligence (XAI) and Supervised Machine Learning-based Algorithms for Prediction of Surface Roughness of Additively Manufactured Polylactic Acid (PLA) Specimens. Applied Mechanics, 4, pp. 668-698. doi:https://doi.org/10.3390/applmech4020034