Neural Networks vs. Traditional Techniques: Differential Equation Approaches in Quantum Gate Analysis

Romik Sargsyan
Russian-Armenian University
Yerevan, Republic of Armenia
e-mail: romik.sargsyan@student.rau.am

Roman Sahakyan
Russian-Armenian University,
Yerevan, Republic of Armenia
e-mail: roman.sahakyan@student.rau.am

Emil Gazazyan
Institute for Physical Research of the National Academy of Sciences of the Republic of Armenia
Institute for Informatics and Automation Problems of National Academy of Science,
Yerevan, Republic of Armenia.
e-mail: emil@quopt.net

Shushanik Sargsyan
Institute for Informatics and Automation Problems of the National Academy of Sciences of the Republic of Armenia,
Yerevan, Republic of Armenia.
e-mail: shush_sarg@mail.ru

Sevak Sargsyan
Centre of Advanced Software Technologies of Russian Armenia University,
Yerevan, Republic of Armenia
e-mail: sevak.sargsyan@rau.am

Hrachya Astsatryan
Institute for Informatics and Automation Problems of the National Academy of Sciences of the Republic of Armenia,
Yerevan, Republic of Armenia.
e-mail: hrach@sci.com

https://Doi.org/10.62343/csit.2024.4

***Abstract* — The research evaluates the traditional numerical algorithms and neural network-based methods for solving ordi-nary differential equations. The evaluation covers both mathe-matical and machine learning perspectives. Practical experi-ments are carried out by implementing neural networks and the necessary algorithms using the Julia programming language. These experiments highlight significant advancements in the realm of quantum computing. Additionally, analyzing the effi-ciency metrics of solving differential equations with numerical methods versus neural networks uncovers intriguing results, which vary based on the chosen architecture and network pa-rameters.**

***Keywords* — Neural Network, differential equation, quantum simulation.**

I. INTRODUCTION

Quantum mechanical computing leverages the fundamental principles of quantum mechanics—including superposition, entanglement, tunneling, and annealing—to tackle problems beyond the reach of classical computers [1]. By utilizing quantum bits (qubits) and these quantum phenomena, quantum computers have the potential to revolutionize fields such as material science, cryptography, and optimization with unparalleled efficiency [2, 3]. Notably, quantum computing excels at processing complex differential equations more efficiently than traditional computers, enhancing the numerical solutions for these equations [4]. Integrating neural networks (NN) [5] is pivotal in advancing quantum mechanical computing, as NNs adeptly manage intricate data, optimize parameters, and enhance the precision of quantum simulations.

Neural networks enable cutting-edge technologies to process information at the molecular, atomic, and quantum dot levels. These advancements promise to transform cryptography and computational systems by enabling irreversible data processing, thereby significantly improving information security. Implementing of reversible quantum gates [6] and quantum programmable logic gates [7] supports the development of sophisticated computational architectures, essential for building robust and intricate systems. Quantum gates manipulate qubits through unitary transformations—reversible operations that precisely alter qubit states—crucial for error correction and quantum communication tasks.

A notable advancement in quantum computing is the creation of a three-bit programmable atomic gate on a five-level atom, achieved through the adiabatic transfer of atomic level populations in an M-system [8]. This adiabatic transfer method allows seamless transitions between atomic energy levels while maintaining quantum coherence, a vital requirement for dependable quantum computations. This innovation facilitates the implementation of complex quantum algorithms and the realization of practical quantum technologies. Combined with other non-reversible quantum gates [9], this approach marks a significant progression in the field.

The study focuses on the numerical solutions of non-stationary equations governing the density matrix of this five-level system. Quantum gates play a crucial role in solving complex systems of 25 differential equations, enabling a deeper understanding and modeling of dynamic processes. Specifically, the research examines the relaxation rates of ground state levels in rubidium atomic vapor through fluorescence spectra analysis, a typical regression task [10]. Rubidium vapor is particularly suited for investigating quantum phenomena, atomic clocks, and quantum computing applications. Both linear and nonlinear machine learning (ML) techniques were evaluated for their effectiveness in processing and predicting these physical processes [11]. An optimal regression model was developed, distinguished by its high accuracy and ability to significantly accelerate the modeling of critical functional indicators, thereby enhancing the prediction and analysis of physical phenomena.

The neural network approach for solving differential equations was successfully implemented using TensorFlow [12], addressing both ordinary differential equations (ODE) and partial differential equations (PDE) that do not involve complex numbers. The method demonstrated exceptional convergence, highlighting its advantages and significantly reducing errors compared to traditional techniques such as Runge-Kutta and finite difference methods [13, 14]. Additionally, this neural approach offers superior numerical stability, avoiding the limitations imposed by the Courant-Friedrichs-Lewy condition [15], and can effectively solve PDEs on very coarse grids.

Among traditional numerical methods [4], neural networks [5, 16, 17] stand out as powerful tools for solving differential equations. They are capable of modeling complex dependencies and identifying patterns within data. Applying neural networks to differential equation solving opens new avenues in dynamic system analysis, enabling highly accurate approximate solutions and achieving these results with significantly reduced computational time compared to conventional numerical methods.

The article seeks to assess traditional numerical methods and NN approaches that handle complex numbers for solving ordinary differential equations within a quantum two-level system in quantum mechanics. By comparing the effectiveness of these numerical techniques and NNs, the study highlights notable differences in computational accuracy, emphasizing the potential of neural networks in addressing intricate mathematical challenges, especially in quantum computing. These results carry significant implications for future quantum physics research, particularly in simulating quantum computations where conventional methods might restrict the speed and precision of data processing.

II. NEURAL NETWORKS FOR SOLVING ORDINARY DIFFERENTIAL EQUATION

The study concentrates on the design of the neural network architecture, including activation functions, optimization algorithms, and training strategies, to ensure precise and effective system dynamics modeling. The network architecture is meticulously crafted to manage the complexities of the quantum two-level system under investigation, as depicted in Fig. 1. It comprises an input layer, two hidden layers, and an output layer. The input layer features a single neuron, representing the one-dimensional time input characteristic of dynamic systems. Each hidden layer contains 64 neurons, providing ample capacity to handle the network's information processing demands. The output layer consists of four neurons, each directly corresponding to the four components of the system's state vector, ensuring a clear and direct relationship between the network's output and the physical states of the quantum system.

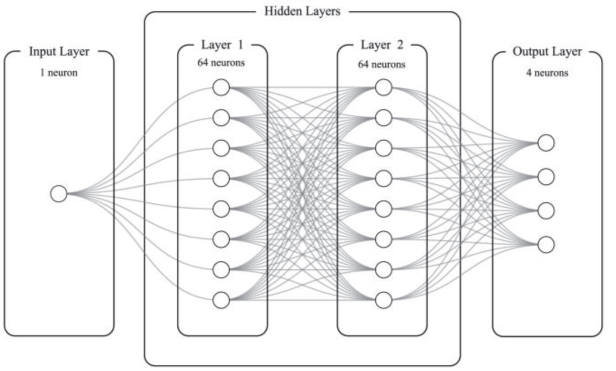

Fig. 1: NN Model Architecture

The hidden layers use a complex sigmoid function for activation, which is crucial for handling complex-valued inputs and outputs. By integrating the real and imaginary components of complex numbers, this function allows the network to accurately simulate the subtle and intricate dynamics characteristic of quantum mechanical systems.

Network optimization employs the Adam optimizer [19, 20], renowned for efficiently navigating complex optimization landscapes and achieving rapid convergence. The learning rate is set at 0.01 to balance the speed and stability of the training process.

The training regime adopts a stochastic approach, introducing randomness in selecting data points to robustly mitigate the risk of overfitting. This randomness prevents the model from excessively memorizing the training data, enhancing its generalization capabilities and ensuring adaptability to various scenarios within the quantum system. The training dataset comprises 300 carefully selected points, balancing computational efficiency with the complexity required to model the dynamic behaviors adequately. Additionally, 30,000 boundary condition points are included to ensure the network rigorously adheres to the physical constraints and boundary conditions essential for obtaining valid and meaningful solutions.

Network parameters are initialized using the kaiming_normal() function [19, 20], which scales the weights based on a calculated standard deviation. This method maintains consistent variances across neuron inputs and outputs, preserving the integrity of signal propagation through the network layers. Such initialization is particularly effective for networks handling complex values, as it helps avoid issues like vanishing or exploding gradients.

Before training, additional setup steps are performed using Lux.setup(rng, chain) [19, 20]. These steps involve crucial tasks such as parameter initialization and system configuration, ensuring that the network is optimally prepared for the training phase. This setup is vital for aligning the network's operational parameters with its architectural and strategic goals, thereby establishing a solid foundation for efficient learning and accurate performance in complex simulation tasks.

In addressing complex-valued Ordinary Differential Equations (ODEs), the study employs the NNODE approach. This method leverages the flexibility and adaptability of neural networks to approximate solutions to differential equations that are otherwise challenging for traditional numerical solvers. Integrating neural networks into the framework of differential equations offers a promising avenue for tackling problems involving complex-valued systems and intricate dynamics.

The NNODE method utilizes a neural network architecture, comprising multiple layers, to represent the solution to an ODE. The network receives the independent variables of the ODE (e.g., time) as inputs and generates predictions for the dependent variables (e.g., state vectors). A loss function, derived from the given differential equation, is defined to train the network and optimize its parameters (weights and biases). This loss function assesses how well the neural network's output satisfies the ODE across a range of input values.

The essence of the NNODE approach lies in its iterative training process, wherein the network minimizes the discrepancy between its predicted solutions and the true solutions of the ODE. The training involves three key components: the optimization algorithm [21], the loss function, and the training strategy. Various strategies can be employed to determine the points at which the ODE is evaluated during training. Techniques such as stochastic training introduce randomness in the selection process, thereby enhancing the network's generalization capabilities and preventing overfitting to specific data points or patterns.

The L2 loss function (1), commonly used in training neural networks with the NeuralPDE.jl package, is calculated based on the squared L2 norm between the network's predictions and the target values:

$$L2 = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \qquad (1)$$

where y_i is the target value, $\hat{y}_i$ is the corresponding predicted value, and n is the number of observations.

III. NEURAL NETWORKS FOR QUANTUM SIMULATIONS

The study investigates the dynamics of a two-level quantum system, which is distinguished by its capacity to exist in a superposition of two distinct quantum states. These systems, referred to as qubits, are fundamental for comprehending the principles of quantum mechanics. To analyze their behavior, mathematical methods based on linear differential equations in two-dimensional spaces are employed. Specifically, the optical Bloch equations [22, 23] are utilized to model the dynamics of atoms and quantum bits under various external field interactions. These equations incorporate the effects of field interactions alongside quantum mechanical properties such as coherence and superposition.

The optical Bloch equations governing the system are expressed as follows:

$$\begin{cases} \frac{d\rho_{11}}{dt} = \frac{i\Omega}{2}(\rho_{12} - \rho_{21}) + \Gamma\rho_{22} \\ \frac{d\rho_{22}}{dt} = -\frac{i\Omega}{2}(\rho_{12} - \rho_{21}) - \Gamma\rho_{22} \\ \frac{d\rho_{12}}{dt} = -(\gamma + i\Delta)\rho_{12} - \frac{i\Omega}{2}(\rho_{22} - \rho_{11}) \\ \frac{d\rho_{21}}{dt} = -(\gamma - i\Delta)\rho_{21} + \frac{i\Omega}{2}(\rho_{22} - \rho_{11}) \end{cases} \qquad (2)$$

where the matrix elements ρ_{ij} represent the density operator of the system, Ω denotes the Rabi frequency, Γ is the natural decay rate of the excited state, indicating how swiftly the system returns to its ground state in the absence of external disturbances. γ describes the transverse decay rate, capturing the effects of environmental noise and interactions that cause the degradation of quantum information stored within the system. Δ is the detuning parameter, representing the difference between the external field frequency and the natural frequency of the quantum transition.

Tables 1, 2, and 3 provide detailed data on the values of the loss function (L2) and correlation coefficients when solving differential equations using neural networks (NN) for various parameters Δ and Ω. The loss function values reflect the error between predicted and actual solutions, while the correlation coefficients measure the similarity between the predicted and actual quantum states. Varying the values of Δ and Ω allows for the examination of how different parameters affect the accuracy and correlation of neural networks in quantum simulations.

	Δ = -1	*Δ* = 0	*Δ* = 1
Loss function (L2)(2)	0.0000875	0.000805	0.0000869
The correlation coefficients between ρ_{11}^{n} and ρ_{11}^{p}	0.99829	0.9993768	0.9990431
The correlation coefficients between ρ_{22}^{n} and ρ_{22}^{p}	0.99834	0.9994245	0.9990781

Table 1 For Ω=3Γ and Δ loss function and correlation

	Δ = -1	*Δ* = 0	*Δ* = 1
Loss function (L2)(2)	0.0003614	0.00028854	0.000315279
The correlation coefficients between ρ_{11}^{n} and ρ_{11}^{p}	0.99613571	0.99775968	0.996612334
The correlation coefficients between ρ_{22}^{n} and ρ_{22}^{p}	0.99637116	0.99783555	0.996645611

Table 2 Ω=4Γ and Δ loss function and correlation

	Δ = -1	*Δ* = 0	*Δ* = 1
Loss function (L2)(2)	0.001605311	0.001269584	0.0012727
The correlation coefficients between ρ_{11}^{n} and ρ_{11}^{p}	0.989457997	0.990168838	0.9909499
The correlation coefficients between ρ_{22}^{n} and ρ_{22}^{p}	0.989868546	0.990391949	0.9910183

Table 3 Ω=5Γ and Δ loss function and correlation

These data enable the assessment of the accuracy and efficiency of NN methods compared to traditional numerical methods, such as the Runge-Kutta method. The loss function (L2) is used to measure the accuracy of the neural network's predictions relative to benchmark values, where low values of the loss function indicate high prediction accuracy. In the presented tables, the loss function values vary depending on the parameters Δ and Ω. Correlation coefficients are used to evaluate the degree of correspondence between the values obtained using NNs (ρ_{11}^{p} and ρ_{22}^{p}) and the values obtained by traditional numerical methods (ρ_{11}^{n} and ρ_{22}^{n}). High correlation coefficients indicate a strong relationship between the two data sets and confirm that neural networks can accurately model the behavior of complex systems.

The obtained data demonstrate the high accuracy of NN methods in solving differential equations and show a strong correspondence with the results of traditional numerical methods. This indicates that neural networks are a reliable tool for modeling complex quantum systems, providing high accuracy and efficiency compared to classical approaches.

IV. CONCLUSION

This work presents an in-depth comparative analysis of algorithms from traditional numerical methods and methods using NNs that handle complex numbers and the choice of activation functions for solving ordinary differential equations in quantum computing. Experiments were conducted during the study, demonstrating the efficiency of both approaches depending on the characteristics of the architecture and network parameters. The efficiency metrics of numerical methods and NNs indicate significant differences in accuracy and computational speed, highlighting the potential of NNs in solving complex mathematical These findings could have substantial implications for future research in the fields of physics and engineering, where traditional problems, especially in the field of quantum computing. Methods may limit the speed and accuracy of data processing.

ACKNOWLEDGMENT

The experiments utilized computational resources from the Armenian cloud infrastructure [24]. Armenian cloud provided access to nearly 700 vCPUs. The Higher Education and Science Committee of RA supported the work in the frames of projects N 1-6/IPR and IIAP 1-8/24-I/IIAP

REFERENCES

1. R. P. Feynman, “Quantum mechanical computers,” Foundations of Physics, vol. 16, pp. 507–531, Jun 1986.
2. U. R. Karpuzcu, ‘Special Issue on Quantum Computing’, *IEEE Micro*, vol. 41, no. 5, pp. 6–7, 2021.
3. D. Richart, W. Laskowski, Y. Fischer, and H. Weinfurter, ‘Experimental analysis of qudit entangled states using the time-energy degree of freedom’, in 2013 Conference on Lasers & Electro-Optics Europe & International Quantum Electronics Conference CLEO EUROPE/IQEC, 2013, pp. 1–1.
4. C. Rackauckas and Q. Nie, “DifferentialEquations.jl–a performant and feature-rich ecosystem for solving differential equations in Julia,” Journal of Open Research Software, vol. 5, no. 1, 2017.
5. K. Zubov, Z. McCarthy, Y. Ma, F. Calisto, V. Pagliarino, S. Azeglio,L. Bottero, E. Luj´an, V. Sulzer, A. Bharambe, N. Vinchhi, K. Balakrishnan, D. Upadhyay, and C. Rackauckas, “Neuralpde: Automating physics-informed neural networks (pinns) with error approximations,” 2021.
6. G. Grigoryan, V. Chaltykyan, E. Gazazyan, and O. Tikhova, “All-optical four-bit toffoli gate with possible implementation in solids,” Proc SPIE, vol. 8772, 05 2013.
7. E. A. Gazazyan, G. G. Grigoryan, V. O. Chaltykyan, and D. Schraft, “Implementation of all-optical toffoli gate in λ-systems,” Journal of Contemporary Physics (Armenian Academy of Sciences), vol. 47, pp. 216–221,Sep 2012.
8. A. Y. Aleksanyan and E. A. Gazazyan, “Realization of the programmable logical atomic gate,” Journal of Contemporary Physics (Armenian Academy of Sciences), vol. 53, pp. 205–211, Jul 2018.
9. E. A. Pogosyan and E. A. Gazazyan, “Programmable quantum gate operations using qutrit quantum dots,” Optical Memory and Neural Networks, vol. 32, pp. S396–S401, Dec 2023.
10. A. Aleksanyan, S. Shmavonyan, E. Gazazyan, A. Khanbekyan, H. Azizbekyan, M. Movsisyan, and A. Papoyan, “Fluorescence of rubidium vapor in a transient interaction regime,” Journal of the Optical Society of America B, vol. 37, p. 203, 12 2019.
11. A. A. Sargsyan, A. Y. Aleksanyan, S. A. Petrosyan, E. A. Gazazyan, A. V. Papoyan, and H. V. Astsatryan, “Prediction of atomic ground state relaxation rate from fluorescence spectra using machine learning,” Journal of Contemporary Physics (Armenian Academy of Sciences), vol. 56, pp. 285–290, Oct 2021.
12. F. Sim, E. Budiarto, and R. Rusyadi, “Comparison and analysis of neural solver methods for differential equations in physical systems,” ELKHA: Jurnal Teknik Elektro, vol. 13, no. 2, pp. 134–140, 2021.
13. Ascher, Uri M.; Petzold, Linda R. (1998), Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations, Philadelphia: Society for Industrial and Applied Mathematics, ISBN 978-0-89871-412-8.
14. Atkinson, Kendall A. (1989), An Introduction to Numerical Analysis (2nd ed.), New York: John Wiley & Sons, ISBN 978-0-471-50023-0.
15. R. Courant, K. Friedrichs, and H. Lewy, “ ¨Uber die partiellen differen-zengleichungen der mathematischen physik,” Mathematische Annalen, vol. 100, pp. 32–74, Dec 1928.
16. K. Zubov, Z. McCarthy, Y. Ma, F. Calisto, V. Pagliarino, S. Azeglio, L. Bottero, E. Luj´an, V. Sulzer, A. Bharambe, et al., “Neuralpde: Automating physics-informed neural networks (pinns) with error approximations,” arXiv preprint arXiv:2107.09443, 2021.
17. E. Shi and C. Xu, “A comparative investigation of neural networks insolving differential equations,” Journal of Algorithms & Computational Technology, vol. 15, p. 1748302621998605, 2021.
18. J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, “Julia: A fresh approach to numerical computing,” SIAM Review, vol. 59, no. 1, pp. 65–98, 2017.
19. M. Innes, “Flux: Elegant Machine Learning with Julia”, Journal of Open Source Software, 2018.
20. M. Innes et al., ‘Fashionable Modelling with Flux’, CoRR, vol. abs/1811.01457, 2018.
21. V. K. Dixit and C. Rackauckas, “Optimization.jl: A unified optimization package,” Mar. 2023.
22. C. J. Foot, Atomic physics. Oxford: Oxford University Press, 2007.
23. R. Loudon, The Quantum Theory of Light. Oxford science publications, Clarendon Press, 1973.
24. [24] D. Petrosyan, H. Astsatryan. “Serverless high-performance computing over cloud." Cybernetics and Information Technologies vol. 22, no. 3, pp. 82-92, 2022.