

Quantum Neural ODEs: Towards Exponential Quantum Advantage in Inference and Learning

Abhinav Muraleedharan^{*1} and Nathan Wiebe^{†1}

¹University of Toronto

The forward pass of deep neural networks involve sequential nonlinear transformations, and the models that now dominate the field are very large in size, making both inference and training extraordinarily expensive. A natural question is whether we can obtain quantum advantage for inference and training of large deep learning models. Indeed, by Bennett’s reversible computation construction, any classical computation can be implemented on a quantum computer with only polynomial overhead. However, such reductions do not generally yield exponential quantum advantages, since the quantum resources required remain proportional to the size of the original classical computation. Consequently, directly quantizing standard feedforward neural networks does not by itself circumvent the scaling limitations of classical deep learning. For exponential quantum advantage, the nonlinear transformations has to act on quantum amplitudes, and these transformations are generally hard to implement on quantum computers since it may lead to violations of many no-go theorems. Therefore, quantum advantage, if exists would only be restricted to classical neural networks in certain parameter regimes, and in this work, we characterize exponentially large classical neural networks that can be efficiently implemented on quantum computers.

Our starting point is the observation, central to the *Neural ODE* viewpoint [1], that a deep network can be viewed as a dynamical system observed at the end of its evolution. A residual network $\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{f}(\mathbf{x}_t, \theta_t)$ can be seen as a Euler discretization of an ordinary differential equation $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t, \theta)$, in which the depth of the network is the elapsed time and the output is the state at the final time. Once the forward pass is seen as *solving a differential equation*, it becomes a simulation problem—and there is by now a mature line of quantum algorithms for exactly this, from Hamiltonian simulation and linear-system solvers [2, 3, 4, 5, 6] to more recent methods that handle classes of *nonlinear* ODEs through Carleman linearization and other techniques [7, 8, 9, 10, 11, 12, 13, 14, 15]. These developments suggest a route to quantum machine learning that accelerates the continuous dynamics underlying a residual neural network. In this work, we present *Quantum Neural ODEs*, the quantum analog of classical neural ODEs. Specifically, we present quantum algorithms to run the forward pass of an exponentially large continuous-time network and to learn such a network from data, both using only polynomially many quantum resources. Along the way we show the resulting simulation problem is BQP-complete, so it gives strong evidence that sampling from the outputs of exponentially large, quantumly simulable neural networks is classically hard.

Running the forward pass. A wide range of Neural ODEs can be written as a quadratic equation $\dot{\mathbf{x}} = \mathbf{A}_1\mathbf{x} + \mathbf{A}_2(\mathbf{x} \otimes \mathbf{x})$, and this quadratic form is more expressive than it first appears. Networks with ReLU activations can be approximated using polynomials, for a defined range and with the introduction of a few auxiliary coordinates, and can be embedded into a slightly larger quadratic ODE system. The obstacle to running this on a quantum computer is that quantum evolution is *linear* while the equation is not. We remove the obstacle by applying

^{*}abhi@cs.toronto.edu

[†]nawiebe@cs.toronto.edu

Carleman linearization Techniques [9, 14]: introducing the tensor powers $\mathbf{x}, \mathbf{x}^{\otimes 2}, \mathbf{x}^{\otimes 3}, \dots$ as auxiliary variables turns one nonlinear differential equation into an infinite dimensional *linear* ODE. When the nonlinear ODE satisfy Carlemann convergence criteria [14, 16], we can truncate this infinite dimensional system at an order $k = \mathcal{O}(\log(T/\epsilon))$, and simulate the system using quantum algorithms for solving linear differential equations. In the special case when the linear term is anti-hermitian, a quantum state proportional to the truncated Carleman vector can be prepared with $\mathcal{O}(\alpha k^2 T + k \log(\|\mathbf{H}_2\| k^2 T/\epsilon))$ queries to block encodings of the coefficient matrices and $\mathcal{O}(k \log(Nk))$ qubits, where α captures the sparsity and magnitude of the coefficients [15]. The cost is polynomial in the natural parameters and, crucially, only *polylogarithmic in the dimension of hidden state* N .

While quantum algorithms for solving differential equations focus on extracting useful physical quantities from the prepared Carlemann State, in machine learning applications such as classification, we want to *sample* the label, i.e. sample from the softmax over the output logits. We give a subroutine that prepares a state whose amplitudes are the softmax weights $e^{\beta h_i(T)/2}$. Since the Carleman state already carries all the tensor powers $\mathbf{x}(T)^{\otimes i}$, and the componentwise exponential is just a particular weighted sum of those powers (its Taylor series), so a single non-unitary, block-encoded operator assembles the softmax state directly from the prepared Carleman state. We prove that truncating this expansion at order $k' = \mathcal{O}(\log(\sqrt{D}/\epsilon))$ gives accuracy ϵ , and that the amplitude-amplification overhead of projecting onto the successful branch is $\sqrt{D/\mathcal{Z}(T)} = 1/\sqrt{\langle e^{\beta h(T)} \rangle}$ —the inverse square root of the mean softmax weight. Switching off the nonlinear term reduces the forward pass to ordinary sparse Hamiltonian simulation, which is *BQP-complete*. Hence, simulating Neural ODE dynamics captures the full power of quantum computation.

Learning the network from data. A fast forward pass is only useful if we can also *train* the model, and here our approach departs sharply from variational quantum machine learning. The dominant paradigm parametrizes a circuit and tunes it in a measure–update–repeat loop, which suffers from barren plateaus and from the cost of reading classical information back out at every step. We avoid the loop by writing each learning problem as a regression with a *closed-form* solution, carried out directly with quantum linear algebra, leaving the learned parameters in a resource state—closely related to the pseudo-Choi states used in recent Hamiltonian-learning work [17]—from which the coefficients of the generator can be read off by shadow tomography [18] or quantum mean estimation [19].

The cleanest case is a unitary teacher, $\mathbf{y}_i = e^{-i\mathbf{H}t} \mathbf{x}_i$, with $\mathbf{H} = \sum_{j=1}^L \theta_j \mathbf{P}_j$ in a known k -local Pauli basis. Fitting the best unitary to the data is the classic orthogonal Procrustes problem [20], solved by a singular value decomposition of the data correlation matrix $\mathbf{X}\mathbf{Y}^\dagger$; we realize this on the quantum computer using the singular value transformation [21] followed by a matrix logarithm to recover the Hamiltonian, preparing the resource state in $\mathcal{O}(\kappa(\mathbf{X}\mathbf{Y}^\dagger) \log^2(1/\epsilon))$ queries. Reading out the L Pauli coefficients then costs $\tilde{\mathcal{O}}(L/\epsilon^2)$ by shadow tomography [18], or $\tilde{\mathcal{O}}(\sqrt{L}/\epsilon)$ by quantum mean estimation [17]. Relaxing the teacher to a general linear generator $\mathbf{y}_i = e^{\mathbf{A}t} \mathbf{x}_i$ replaces the unitary fit with an ordinary least-squares fit $\widehat{\mathbf{M}} = \mathbf{Y}\mathbf{X}^+$ followed by $\mathbf{A} = \frac{1}{t} \log \widehat{\mathbf{M}}$. The technical enabler is a new subroutine which we introduce that block-encodes the logarithm of a general matrix (any matrix whose eigenvalues avoid the negative real axis) through the integral identity $\log \mathbf{M} = (\mathbf{M} - \mathbf{I}) \int_0^1 [t(\mathbf{M} - \mathbf{I}) + \mathbf{I}]^{-1} dt$, evaluated by Gauss–Legendre quadrature and quantum matrix inversion in $\tilde{\mathcal{O}}(\alpha_M \kappa_{\max}^2)$ queries; because the coefficients here may be complex, we read out the real and imaginary parts of each Pauli coefficient separately, again at $\tilde{\mathcal{O}}(M/\epsilon^2)$ or $\tilde{\mathcal{O}}(\sqrt{M}/\epsilon)$ cost.

The richest and most interesting case is a genuinely *nonlinear* teacher, where the data are produced by the quadratic ODE itself and we must learn both $\mathbf{A}_1$ and $\mathbf{A}_2$. The natural move is to lift the data by the same Carleman construction—stacking $\bar{\mathbf{x}} = [\mathbf{x}, \mathbf{x}^{\otimes 2}, \dots, \mathbf{x}^{\otimes k}]$ —so that the

lifted pairs are related by a single *linear* propagator $\bar{\mathbf{y}} \approx e^{\bar{\mathbf{A}}t} \bar{\mathbf{x}}$, and then reuse the linear learner. The subtlety, and the crux of this part of the work, is that $\bar{\mathbf{A}}$ is *not* a free matrix: it must be block upper-bidiagonal, its diagonal blocks are fixed tensor-lifts of $\mathbf{A}_1$, and its upper blocks are fixed tensor-lifts of $\mathbf{A}_2$, so the only independent parameters sit in the first block row and everything else is determined algebraically. A learner that ignored this structure would return a generator not corresponding to an actual quadratic ODE and would have a large generalization error. We therefore turn the training into a *constrained* regression, encoding the Carleman consistency relations as a single linear condition $\mathbf{\Pi} \text{vec}(\bar{\mathbf{A}}) = \mathbf{0}$ on the unknown generator (with $\|\mathbf{\Pi}\| = \mathcal{O}(k^{3/2})$). This constraint can be imposed by appending it to the regression, yielding an augmented least-squares problem, which can be solved using QSVT based techniques.

We work this out in two regimes. When the data are sampled over short intervals, the propagator linearizes as $e^{\bar{\mathbf{A}}\delta t} \approx \mathbf{I} + \delta t \bar{\mathbf{A}}$, and after vectorization the task collapses to an ordinary least-squares system $\mathbf{L}\mathbf{a} = \mathbf{y}$ in $\mathbf{a} = \text{vec}(\bar{\mathbf{A}})$; appending the constraints gives $\tilde{\mathbf{L}} = [\mathbf{L}; \mathbf{\Pi}]$ with closed-form solution $\mathbf{a}^* = \tilde{\mathbf{L}}^+ \mathbf{y}$. Here we also prove a generalization bound: under Gaussian observation noise of variance σ^2 , the expected parameter error obeys $\mathbb{E}\|\tilde{\mathbf{a}} - \mathbf{a}^*\|_2 \leq \sigma \sigma_{\max}(\mathbf{L}) \sqrt{r}/\sigma_{\min}(\tilde{\mathbf{L}})^2$, so accuracy is controlled by the conditioning of the augmented system. For long horizons the first-order picture is insufficient, so we expand the propagator to order m and use the identity $\text{vec}(\bar{\mathbf{A}}^p) = \mathbf{C}_p \text{vec}(\bar{\mathbf{A}})^{\otimes p}$ to re-express each power as a linear map on tensor powers of the unknown, producing one large augmented system $\tilde{\mathbf{F}}\tilde{\mathbf{a}} = \tilde{\mathbf{b}}$ that stacks the data equations against the Carleman constraints applied at every tensor level, with solution $\tilde{\mathbf{a}}^* = \tilde{\mathbf{F}}^+ \tilde{\mathbf{b}}$. Preparing this solution as a quantum resource state costs $\mathcal{O}(\alpha_F \kappa_F^2 \log(\kappa_F/\epsilon))$ queries and only $\mathcal{O}(m \log n)$ qubits, so the whole problem hinges on the condition number κ_F , which our main technical estimate bounds *independently of the ambient dimension*:

$$\kappa_F \leq \frac{\sqrt{2} \sqrt{\sigma_{\max}^2(\mathbf{X}) \Phi_{\Sigma}^2 + \pi_{\max}^2}}{\lambda}, \quad \Phi_{\Sigma}^2 = \sum_{p=1}^m \frac{\|\mathbf{C}_p\|^2}{(p!)^2},$$

where the numerator collects the spread of the data $\sigma_{\max}(\mathbf{X})$ and the size of the Carleman lift (Φ_{Σ} and $\pi_{\max} = \|\mathbf{\Pi}_{\text{blk}}\|$), and λ is the weight corresponding to a regularizing term in the objective. After preparing the quantum state proportional to $\tilde{\mathbf{a}}^*$, we apply Shadow Tomography techniques to extract Pauli coefficients corresponding to $\mathbf{A}_1$, and $\mathbf{A}_2$.

Our results connect three areas—deep learning, nonlinear dynamical systems, and quantum algorithms—through one idea: treat exponentially large classical neural networks as dynamical systems and apply quantum algorithms to simulate the "forward pass". Doing so lets us run the forward pass of a certain class of exponentially wide continuous-time networks in cost polylogarithmic in its size, sample from its output efficiently, and learn its parameters without a variational loop. Our results suggest a setting where exponential quantum advantage in machine learning may be provably attainable; they complement recent quantum approaches to accelerating neural-network inference from a different angle [22].

References

- [1] Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David Duvenaud. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, 2018.
- [2] Seth Lloyd. Universal quantum simulators. *Science*, 273(5278):1073–1078, 1996.
- [3] Aram W. Harrow, Avinandan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103(15):150502, 2009.
- [4] Dominic W. Berry, Andrew M. Childs, and Robin Kothari. Hamiltonian simulation with nearly optimal dependence on all parameters. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 792–809, 2015.

- [5] Dominic W. Berry, Andrew M. Childs, Aaron Ostrander, and Guoming Wang. Quantum algorithm for linear differential equations with exponentially improved dependence on precision. *Communications in Mathematical Physics*, 356(3):1057–1081, 2017.
- [6] Andrew M. Childs, Robin Kothari, and Rolando D. Somma. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM Journal on Computing*, 46(6):1920–1950, 2017.
- [7] Sarah K. Leyton and Tobias J. Osborne. A quantum algorithm to solve nonlinear differential equations. *arXiv preprint*, 2008.
- [8] Seth Lloyd, Giacomo De Palma, Can Gokler, Bobak Kiani, Zi-Wen Liu, Milad Marvian, Felix Tennie, and Tim Palmer. Quantum algorithm for nonlinear differential equations. *arXiv preprint*, 2020.
- [9] Jin-Peng Liu, Herman Øie Kolden, Hari K. Krovi, Nuno F. Loureiro, Konstantina Trivisa, and Andrew M. Childs. Efficient quantum algorithm for dissipative nonlinear differential equations. *Proceedings of the National Academy of Sciences*, 118(35):e2026805118, 2021.
- [10] Hari Krovi. Improved quantum algorithms for linear and nonlinear differential equations. *Quantum*, 7:913, 2023.
- [11] Jin-Peng Liu, Dong An, Di Fang, Jiasu Wang, Guang Hao Low, and Stephen Jordan. Efficient quantum algorithm for nonlinear reaction–diffusion equations and energy estimation. *Communications in Mathematical Physics*, 404(2):963–1020, 2023.
- [12] Pedro C. S. Costa, Philipp Schleich, Mauro E. S. Morales, and Dominic W. Berry. Further improving quantum algorithms for nonlinear differential equations via higher-order methods and rescaling. *npj Quantum Information*, 11(1):40, 2025.
- [13] Hsuan-Cheng Wu, Jingyao Wang, and Xiantao Li. Quantum algorithms for nonlinear dynamics: Revisiting carleman linearization with no dissipative conditions. *SIAM Journal on Scientific Computing*, 47(2):A943–A970, 2025.
- [14] David Jennings, Kamil Korzekwa, Matteo Lostaglio, Andrew T. Sornborger, Yigit Subasi, and Guoming Wang. Quantum algorithms for general nonlinear dynamics based on the carleman embedding. *arXiv preprint*, 2025.
- [15] Abhinav Muraleedharan and Nathan Wiebe. Simulating time dependent and nonlinear classical oscillators through nonlinear schrödingerization. *arXiv preprint*, 2025.
- [16] Marcelo Forets and Amaury Pouly. Explicit error bounds for carleman linearization. *arXiv preprint*, 2017.
- [17] Juan Castañeda and Nathan Wiebe. Hamiltonian learning via shadow tomography of pseudo-choi states. *Quantum*, 9:1700, 2025.
- [18] Hsin-Yuan Huang, Richard Kueng, and John Preskill. Predicting many properties of a quantum system from very few measurements. *Nature Physics*, 16(10):1050–1057, 2020.
- [19] William J. Huggins, Nathan Wiebe, et al. Nearly optimal quantum algorithm for estimating multiple expectation values. *Physical Review Letters*, 129(24):240501, 2022.
- [20] Peter H. Schönemann. A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31(1):1–10, 1966.
- [21] András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe. Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing (STOC)*, pages 193–204, 2019.
- [22] Arthur G Rattew, Po-Wei Huang, Naixu Guo, Lirandë Pira, and Patrick Rebentrost. Accelerating inference for multilayer neural networks with quantum computers. *arXiv preprint arXiv:2510.07195*, 2025.