nature portfolio

# Peer Review File

# Fourier Neural Operators for Learning Dynamics in Quantum Spin Systems

Corresponding Author: Dr Taylor Patti

**This file contains all reviewer reports in order by version, followed by all author rebuttals in order by version.**

Version 0:

Reviewer comments:

Reviewer #1

(Remarks to the Author)
This manuscript reports on the novel use of Fourier neural operators (FNOs), a neural network architecture utilizing spectral convolutions, to learn solutions to the time-dependent Schrödinger equation. Authors explore several different network architectures, which they use to extrapolate the time evolution of either full wavefunctions, or a set of observables, into times well beyond that used in training. This is demonstrated in 1D spin systems, which have fairly general applicability. The results indicate the potential for high fidelities, with decreased computational complexity in comparison to other approaches. Of particular interest is the extrapolation of observables to later times, as this more clearly provides a path forward to systems where exact simulation is too expensive, and to experimental systems where the full wavefunctions are inaccessible.

Overall the manuscript is well written, and reasonably clear throughout. This work represents a step forward from previous studies, tackling the application of FNOs to many-body quantum systems, with promising results. The demonstration is general enough that researchers in various fields could benefit from this work, and as such, it is impactful. Future directions are also clearly established in the discussion. The manuscript is, however, lacking in some details that I feel are necessary for the reproduction of these results. Furthermore, the error analysis is inconsistent, with regards to the fidelities. In addition to these concerns, there are several more minor points that the authors should consider when revising the manuscript, all detailed below. Assuming the comments below are appropriately addressed by the authors, I recommend this manuscript for publication in Communications Physics.

Major Comments:

1. It is unclear what the lifting layer and ion layers are; are these just typical fully-connected layers? Are there trainable parameters involved in these mappings?
2. Relatedly to (1), it is unclear how the output resolution can be evaluated on finer resolutions than that used in training. This confusion originates from the input and output dimensions relying on the specific choice of discretization, as indicated in Eqs. 9 and 10. From this, it appears that the desired discretization must be decided in advance of the training, so in what way is this approach more general? Its possible that the details of the ion layer will clarify this point already.
3. For these specific demonstrations, the activation functions that were used should be provided in the methods.
4. The precision of nearly all fidelities provided (Figures 2 and 3, the Figure 4 caption, and in the main body, e.g. Section 1B1), and their corresponding errors, are inconsistent. The provided errors have too many digits, they should be rounded to a single digit. Similarly, the results are stated with greater precision than these errors, i.e. they have too many digits. Also, several of the stated fidelities have errors of 0.0, where it seems there should be some statistical uncertainty. In all cases, the fidelities provided should be corrected to precisions informed by the corresponding errors.
5. In the introduction, under "Our Approach", it is stated that data from "...actual quantum computations..." is employed in the training of these FNOs. Based on descriptions in the main text, and methods, this does not appear to be true. As such, either the source and methods for obtaining such data should be clearly stated, or this statement must be revised.

Minor Comments:

1. At the end of the introduction, under "Our approach", authors mention "the U-Net", which has yet to be described at this point in the manuscript.
2. In Section 1B, the random n-particle wavefunctions are described as being sampled from a uniform distribution; this

process is unclear.
3. In Section 1B1, when discussing extrapolation with random inputs, its stated that various time intervals (VTIs) are not necessary for training. While the explanation is intuitive, the VTI results for random inputs are far superior to those without, as seen in Figure 2a; this should be addressed in the manuscript. Furthermore, it is stated in the text that even without VTI, the mean fidelity at extrapolated times is above 94%. It is unclear where this number comes from, seeing as the fidelity in Figure 2 falls below this value. Is it true then only in the case of 4-qubits?
4. The embedding in Section 1B2 is described as using "... the values of two sinusoidal functions applied to the time-grid...". More detail on what these sinusoidal functions are, and how they're chosen, should be provided.
5. There appears to be a typo in the caption to Figure 4, where the extrapolated interval is given as "[2/2 \pi, 7/2 \pi]".

Reviewer #2

(Remarks to the Author)
The paper concerns simulating dynamics of quantum systems using Machine Learning (ML) techniques. In particular, the authors focus in 1D spin system and use the tool of Fourier Neural Operators (FNOs).

While the topic is timely, and ML-derived techniques either are or could be of potentially impact, the manuscript has a number of serious flaws in regards to methodology, execution and presentation, which in my opinion at present should preclude publication. My main concerns are regarding the actual validity of the method for the larger systems, the confusing presentation, and the complete lack of any attempt at a theoretical understanding of the method itself.

Specifically:
1. The system sizes considered are tiny: 4 and 8 qubits. This is important for two reasons:
(A) first with regards to claims and motivations, which prominently mention difficulty in simulating larger systems — at present I don't find any results in the manuscript which give credence to the idea that FNOs allow to address larger quantum systems.
(B) Second, with such small system sizes the dimensionality of the Hilbert space in which the wave function lives is no larger than 256 — comparable in size to the dimensionality of the parameter space of the ML models the authors use (if I'm not mistaken the Authors use e.g. 128 Fourier modes), so the fundamental issue of the "curse of dimensionality" for quantum system is not even really encountered in the numerical experiments in the paper. It is critical to perform larger scale experiments, and in particular to verify the scaling of the fidelity with the increase of the number of the qubits. I notice that the 8-qubit results (still a minute system well within reach of ED methods) are already substantially worse than those for 4 qubits: for what the Authors call "low energy" inputs the fidelity already drops by an order of magnitude (however it is presented, more on that below). I am not at all convinced, based on the presented results, that the methods would work reasonably well even on the simulatable scales of 20-30 qubits, much less on scales the authors mention as their motivations.

2. The paper does not contain any attempt at theoretical understanding how, when, and why the method should work for the dynamics of quantum systems. It feeds some physical data into an ML architecture, including also arbitrary choices (e.g. the number of Fourier modes kept), and observes the final fidelity. The comments about the difficulty os simulating the dynamics of "low energy states" vs. random states seem handwaving and unconvincing. In classical dynamical systems (and in quantum too) there is an understanding that ML-like methods attempting to extrapolate dynamics often manage (at least when they actually make sense) to capture e.g. the leading Koopman modes of the dynamical systems. I would encourage the Authors to compare the results to ones obtained using e.g. Dynamical Mode decomposition (DMD). As it stands, even uf the method would numerically work (and in my opinion it doesn't, at scale) there is understanding or estimates for the error the method makes (other than actually comparing to the exact solution), and no reason to believe it would be capturing "the right thing" when such a comparison would be unavailable.

3. I find the way the data is presented somewhat misleading: we are interested in the simulation fidelity at longer times. Such fidelity always decays in time, essentially monotonically. There is no reason to give in tables the "averaged" fidelity over the whole time interval after training. The only function of this seems to improve the numbers by averaging the infidelity at longer times with the high fidelities at short times, but is otherwise physically meaningless. There are no solid baselines for comparison against any other physical methods given, other than a completely arbitrary, non-standard in physical simulations (and not described in any detail), U-Net machine learning model.

4. The explanation of the technical parts are unclear and the notation is confusing and in parts likely incorrect.
(A) For instance: Eq. (5) states that the object "I" is a complex vector of dimension C^{2*2^n}, and that it contains some object S_0 "which corresponds to the basis states (S)", and then lists a vector of 2^n kets (in Dirac's notation), each of which is presumably a computational basis state, itself of dimension 2^n. This doesn't seem to make sense, even from the point of dimensionality — I assume that the notation is just incorrect and what the Authors really mean is that S_0 is vector of 2^n amplitudes in the computational basis and not of the vectors itself.
(B) Further down in Eq. 7 \psi is used in the Fourier transform formula and referred to as the "state" (when probably the amplitude is meant), without using Dirac's notation, and in the "methods" section the basis vectors or amplitudes are denoted by "\phi_k" instead.

5. Corresponding to the above, the figure 1 also insufficiently reflects the notations in the text (no subscripts of any sort, which do appear in the formulas), and contains parts which seem incorrect (in fig. 1c the vector of kets given as inputs, which manifestly does not have a dimension given in Eq.5, and does not seem to make sense). Fig.1a contains an image which is also highly confusing with some 2D/3D like picture to illustrate a Heisenberg chain in 1D mixing the J_x,J_y,J_z couplings

for the same nearest neighbours along the chains with spatial directions. This is extremely puzzling, to say the least.

6. The explanation of the ”Hamiltonian Observables” section I.C is insufficient. First, the Authors continuously confuse “observables” with “expectation values of observables”. Second, in contradiction to their own statements in the previous section and Fig.2 now they claim that it is the random states which are more difficult to simulate, and they don’t consider the “low energy states “ anymore. This seems arbitrary and unjustified. The observable method is supposed to allow to tackle even larger systems, but they still confine themselves to 4 and 8 qubits. They do not describe the procedure in any detail, and the “methods” section is of not much help either, additionally containing some arbitrary cutoffs.

To sum up, despite the motivations and claims mentioning difficulty in simulating larger systems with other methods, such as e.g. tensor networks, I do not find that the Authors convincingly demonstrate that their results provide an alternative method which is accurate, scales to larger systems and is physically principled. On the contrary, I have very strong doubts about the validity of the method and the quality of the results and their presentation. I find that these are critical faults in the manuscript, and advise a rejection.

Reviewer #3

(Remarks to the Author)
In this work, the authors propose to learn the time evolution of a quantum state using a Fourier Neural Operator. They investigate two architectures, an energy domain one (trained on a single time interval) and time-domain one (trained over multiple time intervals) and show that these architectures can generalize to longer times.

My biggest concern with this work is the scalability and therefore applicability. Although the results seem scientifically accurate, I am not sure if they provide us with enough insight into the efficacy of a Neural Operator approach to quantum dynamics. I therefore would not recommend publication unless the authors can make a strong argument for the scalability of their approach or if they provide a more detailed analysis of the limitations of their approach.

It would be appreciated if the authors provided code to reproduce their results.

Major points:

1.) Where exactly does this method fit in the landscape of numerical approaches? The case where we have access to all 2^n amplitudes seems to have limited applicability. It seems that extrapolating beyond the capabilities of other approaches may be a powerful approach, but I am not sure if this work provides evidence that this can be done given the small scale of the experiments.
2.) “To enhance the learning process, we incorporate a position embedding Embedstate(k) = k/2n into the input channel dimension, where k is the index associated with each basis state. This approach helps the model better understand the relationship between the states and their corresponding energy levels.”

I am very curious about this architectural choice. Is this a common technique in this area? Does the model performance suffer significantly if one does not do this?
3.) How are the low energy states created? How many modes are there?
4.) Why is the performance on random wave functions so good? Should it not be harder to learn unstructurered data from a complexity point of view?
5.) In the same vein. “Importantly, in this study, we focus on using random wave function states to calculate the expectation values of these observables, making the learning and prediction tasks for the FNO maximally general and, as a result, challenging.”
Does this not contradict the results of figure 3, which indicates that the FNO performs better on random states?
6.) I do not understand the ground truth extrapolation. Does this mean that you extrapolate from GT at 5/2pi to 26/10pi and calculate the MRE there? And then from 26/10pi to 27/pi again?
7.) Does the speedup you mention in the conclusion take the training into account? How does the training scale as a function of system size?
8.) Do the authors believe that the an FNO approach could be accurate enough to study new physics? Could an FNO approach, applied at scale provide accurate enough computation of observables to study novel physical phenomenon?

Minor points:

1.) Figure 4 caption, [2/2pi,7/2pi] should be [5/2pi,7/2pi]?

Version 1:

Reviewer comments:

Reviewer #1

(Remarks to the Author)

I am pleased to see that the authors have taken the feedback from all referees, adding necessary revisions to the manuscript; in particular, the addition of the 20 qubit results address the major concerns with scalability.

Aside from the minor comments below, I feel that the revised manuscript has met the necessary acceptance criterion.

Minor comments:
- when discussing the temporal embedding details, authors say “positional embedding” twice. Should this not be “temporal embedding?”

Reviewer #3

(Remarks to the Author)
I want to say that I appreciate the significant improvements the authors made to the manuscript. The exposition is much clearer, and I believe that the results can now be better understood. I also greatly appreciate the open source code to reproduce the experiments.

However, while I appreciate the performance increase over the exact simulation obtained for the larger systems, I am still not completely convinced about the use case of this approach.

It seems that one in most cases of practical interest, one would want to use a more accurate and controlled technique to calculate either the state (e.g. Tensor Networks, Neural quantum states) or observables (e.g. Pauli propagation) for these system sizes, even if it is more expensive. It’s interesting by itself that the dynamics studied here can be compressed reasonably well.

While the authors now propose using measurement data to learn the FNO, this seems like a highly non-trivial extension, since one would now require ive measurements for learning instead of the full states, which arguably contain much less information.

I believe the results in this work are correct, but probably not the most impactful for the field of simulating quantum dynamics. I would therefore recommend publication if the impact is not a major concern for the editors.

Minor: It is still not explained what is meant by “Extrapolated on GT” in figure 4. Does that mean you start with the exact state at 5/2pi?

**Open Access** This Peer Review File is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.
In cases where reviewers are anonymous, credit should be given to 'Anonymous Referee' and the source.
The images or other third party material in this Peer Review File are included in the article’s Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons

license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.
To view a copy of this license, visit https://creativecommons.org/licenses/by/4.0/

Report of the First Referee – COMMSPHYS-24-1031

(0) This manuscript reports on the novel use of Fourier neural operators (FNOs), a neural network architecture utilizing spectral convolutions, to learn solutions to the time-dependent Schrödinger equation. Authors explore several different network architectures, which they use to extrapolate the time evolution of either full wavefunctions, or a set of observables, into times well beyond that used in training. This is demonstrated in 1D spin systems, which have fairly general applicability. The results indicate the potential for high fidelities, with decreased computational complexity in comparison to other approaches. Of particular interest is the extrapolation of observables to later times, as this more clearly provides a path forward to systems where exact simulation is too expensive, and to experimental systems where the full wavefunctions are inaccessible.

Overall the manuscript is well written, and reasonably clear throughout. This work represents a step forward from previous studies, tackling the application of FNOs to many-body quantum systems, with promising results. The demonstration is general enough that researchers in various fields could benefit from this work, and as such, it is impactful. Future directions are also clearly established in the discussion. The manuscript is, however, lacking in some details that I feel are necessary for the reproduction of these results. Furthermore, the error analysis is inconsistent, with regards to the fidelities. In addition to these concerns, there are several more minor points that the authors should consider when revising the manuscript, all detailed below. Assuming the comments below are appropriately addressed by the authors, I recommend this manuscript for publication in Communications Physics.

**Author Comment R1.0:**
We thank the Referee for their careful review of our manuscript, as well as for their conditional acceptance. We also thank them for their thoughtful suggestions and questions. In what follows, we address each of these considerations.

(A) Major Comments:

1. It is unclear what the lifting layer and projection layers are; are these just typical fully-connected layers? Are there trainable parameters involved in these mappings?

**Author Comment R1.A1:**
We thank the Referee for pointing out the lack of clarity regarding the lifting and projection layers. In the revised manuscript, we have rewritten Section 1A2 to explicitly define both mappings and clarify their role in the architecture. We now state that the lifting layer is implemented as a local transformation parameterized by a shallow fully connected neural network that embeds the input data into a higher-dimensional latent channel space. Similarly, we have added the description of the projection layer. Both the lifting and projection layers are now clearly identified as containing trainable parameters and serving as learnable input and output embeddings that interface between the physical data space and the latent space in which the FNO operates.

Original lifting layer discussion in Section 1A2:
*The lifting layer Lift maps the input data **I** with a given number of input channels to a latent function $\boldsymbol{V}_0$ with a (typically) larger number of channels. Here, channels denote the dimensions of the co-domain of the functions $\boldsymbol{V}_\ell$, representing distinct components or features within the latent space. The latent space is an intermediate representation where the data is abstracted into a higher-level form, capturing essential patterns and relationships. The FNO operates within this latent space by applying spectral convolutions.*

New lifting layer discussion in Section 1A2:
*The lifting layer Lift is a local transformation parameterized by a shallow fully connected neural network,*

$$\text{Lift} : \mathbb{R}^{d_i} \to \mathbb{R}^{d_v},$$

*which embeds the input data $\mathbf{I} \in \mathbb{R}^{d_i}$ into a higher-dimensional latent channel space. This produces the initial latent representation*

$$\mathbf{V}_0 = \text{Lift}(\mathbf{I}), \qquad \mathbf{V}_0 \in \mathbb{R}^{d_v},$$

*where typically $d_v > d_i$. Here, channels denote the dimensions of the co-domain of the functions $\boldsymbol{V}_\ell$, representing distinct components or features within the latent space. The latent space is an intermediate representation where the data is abstracted into a higher-level form, capturing essential patterns and relationships. The FNO operates within this latent space by applying spectral convolutions cite [li2020fourier,FNOPracticalReview].*

Original discussion:
*After processing through multiple FNO blocks $F_\ell$, the projection layer Proj maps the latent representation to the output data **O**, which may have one or more channels, depending on the specific architecture used as described in Sections 1B1 and 1B2.*

New discussion:
*After processing through multiple FNO blocks $F_\ell$, the projection layer Proj maps the latent representation back to the output function space.*

$$\text{Proj} : \mathbb{R}^{d_v} \to \mathbb{R}^{d_o},$$

*which yields the predicted output*

$$\mathbf{O} = \mathit{Proj}(\mathbf{V}_L), \qquad \mathbf{O} \in \mathbb{R}^{d_o}.$$

*The output data **O** may have one or more channels, depending on the specific architecture used as described in Sections 1B1 and 1B2.*

2. Relatedly to (1), it is unclear how the output resolution can be evaluated on finer resolutions than that used in training. This confusion originates from the input and output dimensions relying on the specific choice of discretization, as indicated in Eqs. 9 and 10. From this, it appears that the desired discretization must be decided in advance of the training, so in what way is this approach more general? Its possible that the details of the projection layer will clarify this point already.

**Author Comment R1.A2:**
We thank the Referee for raising this important point. We have clarified in Section 1A2 that, although training data are provided on a discrete grid, the FNO learns a function-to-function mapping rather than a mapping between fixed-length vectors. As a result, the trained operator can be evaluated on grids of arbitrary resolution via its Fourier representation, enabling zero-shot super-resolution without retraining.

New discussion in Section IA2:

*Although the input and output are represented on discrete grids during training, the Fourier Neural Operator (FNO) learns a mapping between functions rather than between fixed-length vectors. The lifting and projection layers act pointwise with shared parameters, while the Fourier layers capture global structure in the Fourier space. Together, these components define a continuous, grid-independent operator through its Fourier representation. As a result, the trained operator can be evaluated on discretizations finer than those used in training, enabling zero-shot super-resolution without additional training. This mechanism is closely analogous to spectral interpolation on regular grids, where samples of a periodic, band-limited function defined on an $N$-point uniform grid are transformed to Fourier space, symmetrically zero-padded to a larger spectral resolution $M > N$, and then transformed back via an inverse Fourier transform to obtain the function evaluated on a $M$-point finer grid. This procedure preserves the physically meaningful low-frequency modes without introducing any artificial high-frequency modes, resulting in the evaluation of the same underlying function on a finer grid cite[FNOReviewPaper].*

3. For these specific demonstrations, the activation functions that were used should be provided in the methods.

**Author Comment R1.A3:**
We thank the Referee for noting that the activation function should be explicitly specified. We now specify that we use the Gaussian Error Linear Unit (GELU) for all our experiments.

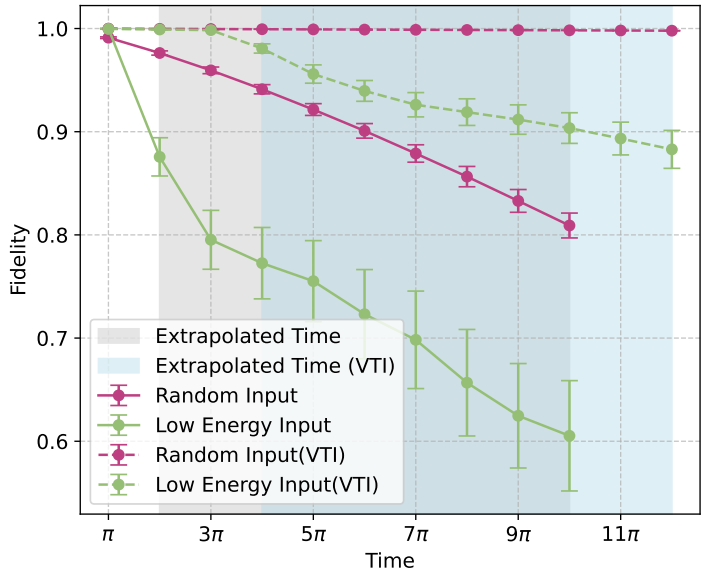
New discussion in Section IA2:
*For all FNO experiments in this work, we employ the Gaussian Error Linear Unit (GELU) as the activation function. We choose GELU because it has been shown to perform well in smooth operator learning tasks and is more flexible than ReLU, allowing small negative outputs and being conducive to more stable training cite[FNOPracticalReview].*

4. The precision of nearly all fidelities provided (Figures 2 and 3, the Figure 4 caption, and in the main body, e.g. Section 1B1), and their corresponding errors, are inconsistent. The provided errors have too many digits, they should be rounded to a single digit. Similarly, the results are stated with greater precision than these errors, i.e. they have too many digits. Also, several of the stated fidelities have errors of 0.0, where it seems there should be some statistical uncertainty. In all cases, the fidelities provided should be corrected to precisions informed by the corresponding errors.

**Author Comment R1.A4:**
We thank the referee for highlighting inconsistencies in the reporting of numerical precision and statistical uncertainties. All fidelities and associated standard deviations in Figures 2–4 and the main text have been revised. Error values are now reported with an appropriate number of significant digits, and cases previously shown with zero uncertainty have been corrected to reflect the underlying statistical variability.

Old Figure 2:

**(a)**




**(b)**

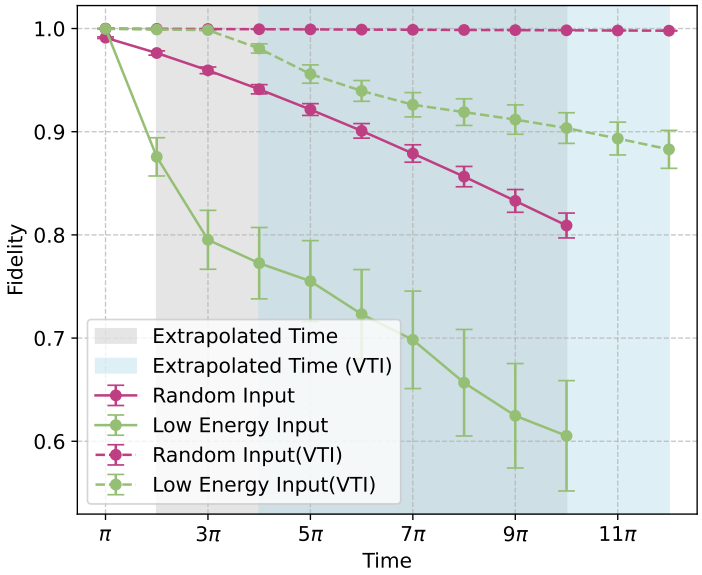
| n | Input | **Fidelity**$[0, \pi]$ | **Fidelity**$[2\pi, 10\pi]$ |
|---|---|---|---|
| 4 | Random | 0.9983 ± 0.0 | 0.9732 ± 2.3e-02 |
| 8 | Random | 0.9920 ± 0.0 | 0.9062 ± 5.8e-02 |

| n | Input | **Fidelity**$[0, 3\pi]$ | **Fidelity**$[4\pi, 12\pi]$ |
|---|---|---|---|
| 4 | Low Energy (VTI) | 0.9999 ± 0.0 | 0.9992 ± 1.0e-03 |
| 8 | Low Energy (VTI) | 0.9996 ± 0.0 | 0.9430 ± 4.2e-02 |

Figure 1: **(a)** Prediction of temporal dynamics in an 8-qubit Heisenberg spin system using the energy-domain architecture in Section 1B1. At time $T = \pi$, the predictions made by the FNO are on unseen data but still within the time range it was trained on. The extrapolated time refers to future time predictions beyond the range on which the FNO was trained. VTI (Various Time Interval) indicates training on multiple time intervals, i.e., $[0, T]$, $[T, 2T]$, and $[2T, 3T]$, instead of training only on the first interval. **(b)** The results show the mean fidelity for time intervals the FNO has seen during training and extrapolated time intervals for systems with 4 and 8 qubits.

New Figure 2:

**(a)**




**(b)**

**Random initial states**

| **n** | $F(\pi)^{\text{train}}$ | $F(2\pi)^{\text{ext}}$ | $F(6\pi)^{\text{ext}}$ | $F(10\pi)^{\text{ext}}$ |
|---|---|---|---|---|
| 4 | $0.9997 \pm 0.0001$ | $0.9991 \pm 0.0003$ | $0.9949 \pm 0.0026$ | $0.9894 \pm 0.0060$ |
| 8 | $0.9911 \pm 0.0009$ | $0.9762 \pm 0.0023$ | $0.9010 \pm 0.0074$ | $0.8101 \pm 0.0129$ |

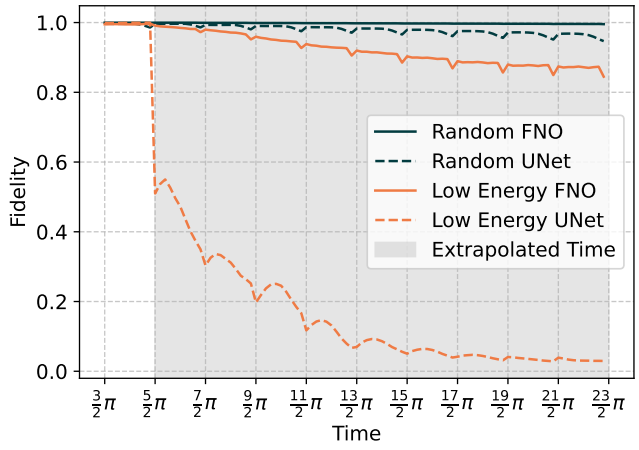
**Low-energy states (VTI)**

| **n** | $F(3\pi)^{\text{train}}$ | $F(4\pi)^{\text{ext}}$ | $F(8\pi)^{\text{ext}}$ | $F(12\pi)^{\text{ext}}$ |
|---|---|---|---|---|
| 4 | $0.9999 \pm 0.0001$ | $0.9999 \pm 0.0001$ | $0.9994 \pm 0.0005$ | $0.9984 \pm 0.0014$ |
| 8 | $0.9984 \pm 0.0002$ | $0.9806 \pm 0.0047$ | $0.9202 \pm 0.0125$ | $0.8836 \pm 0.0184$ |

Figure 2: **(a)** Prediction of temporal dynamics in an 8-qubit Heisenberg spin system using the energy-domain architecture in Section 1B1. At time $T = \pi$, the predictions made by the FNO are on unseen data but still within the time range it was trained on. The extrapolated time refers to future time predictions beyond the range on which the FNO was trained. VTI (Various Time Interval) indicates training on multiple time intervals, i.e., $[0, T]$, $[T, 2T]$, and $[2T, 3T]$, instead of training only on the first interval. **(b)** The results show the fidelity at specific time steps for 4 and 8 qubits, including times within the training range as well as extrapolated future times beyond the training interval. Superscripts indicate whether the particular time step is within the training range (train) or an extrapolated time (ext).

Old Figure 3:

**(a)**




**(b)** FNO

| n | Input Type | Fidelity$[\frac{3}{2}\pi, \frac{5}{2}\pi]$ | Fidelity$[\frac{5}{2}\pi, \frac{23}{2}\pi]$ |
|---|---|---|---|
| 4 | Random | 0.9999 ± 0.0 | 0.9996 ± 3.00e-03 |
| 4 | Low Energy | 0.9999 ± 0.0 | 0.9992 ± 1.00e-03 |
| 8 | Random | 0.9996 ± 2.43e-05 | 0.9978 ± 7.81e-05 |
| 8 | Low Energy | 0.9997 ± 3.83e-05 | 0.9179 ± 9.55e-03 |

**(c)** U-Net

| n | Input Type | Fidelity$[\frac{3}{2}\pi, \frac{5}{2}\pi]$ | Fidelity$[\frac{5}{2}\pi, \frac{23}{2}\pi]$ |
|---|---|---|---|
| 4 | Random | 0.9966 ± 6.52e-05 | 0.9580 ± 8.00e-03 |
| 4 | Low Energy | 0.9999 ± 0.0 | 0.7876 ± 6.10e-02 |
| 8 | Random | 0.9945 ± 5.37e-05 | 0.9790 ± 8.42e-05 |
| 8 | Low Energy | 0.9991 ± 3.59e-05 | 0.1492 ± 6.50e-03 |

Figure 3: **(a)** Prediction of temporal dynamics in an 8-qubit Heisenberg spin system using the time-domain architecture, with FNO and U-Net. The input time interval is denoted as $[0, \frac{3}{2}\pi]$, as illustrated in Figure 1 d), while the extrapolated time refers to future time predictions beyond the trained range. **(b)** The results present the mean fidelity for both the training time (i.e., the time intervals the FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{23}{2}\pi]$ for systems with 4 and 8 qubits, across different input types. **(c)** The results show the mean fidelity using U-Net for similar inputs and time intervals.

New Figure 3:

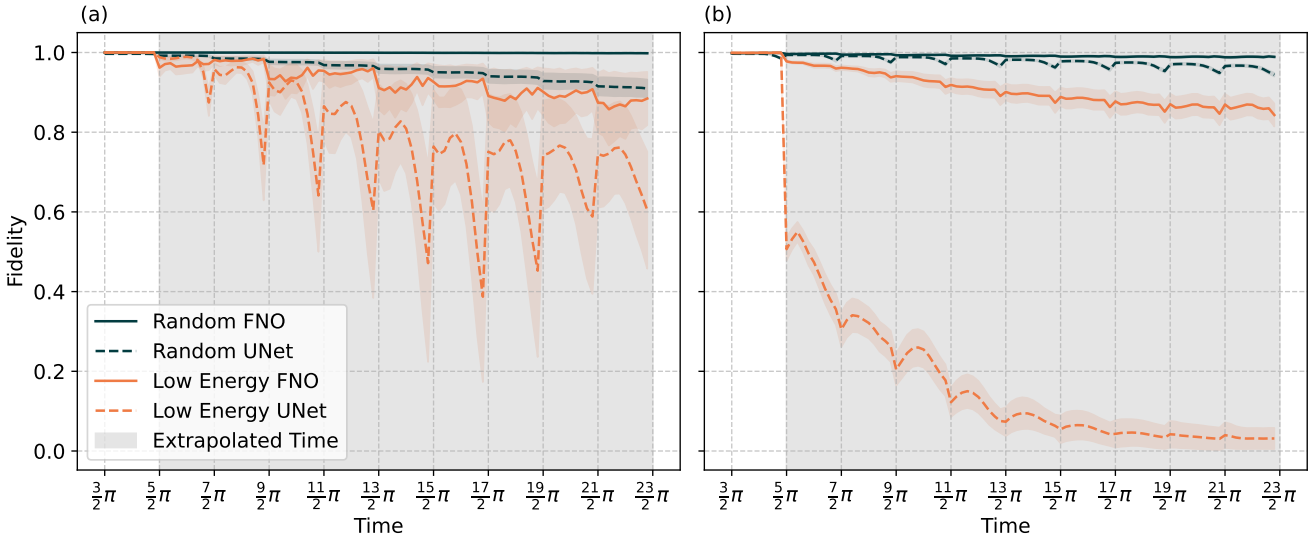



Figure 4: **(a)** Prediction of temporal dynamics in a four-qubit Heisenberg spin system using the time-domain architecture in Section 1B2. The model is trained on the time interval $[0, 3\pi/2]$ as illustrated in Figure 1d, while the extrapolated time refers to future predictions beyond the trained range. Performance for both the random and low-energy wavefunction initial states using the FNO is benchmarked against a U-Net baseline. The shaded regions around each curve indicate one standard deviation. The FNO accurately captures the underlying operator and extrapolates reliably to later times, while the U-Net exhibits poor extrapolation, especially for low-energy input states. **(b)** Corresponding results for an eight-qubit Heisenberg spin system.

Old Figure 4:

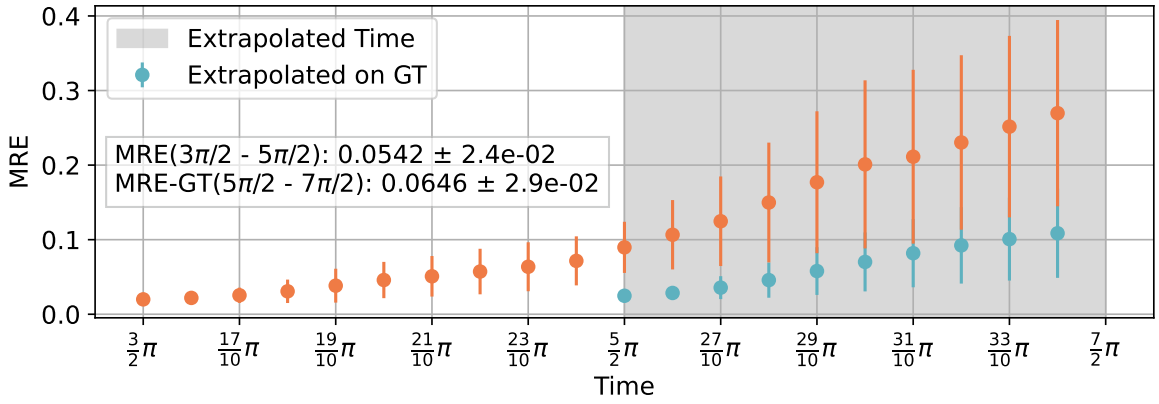
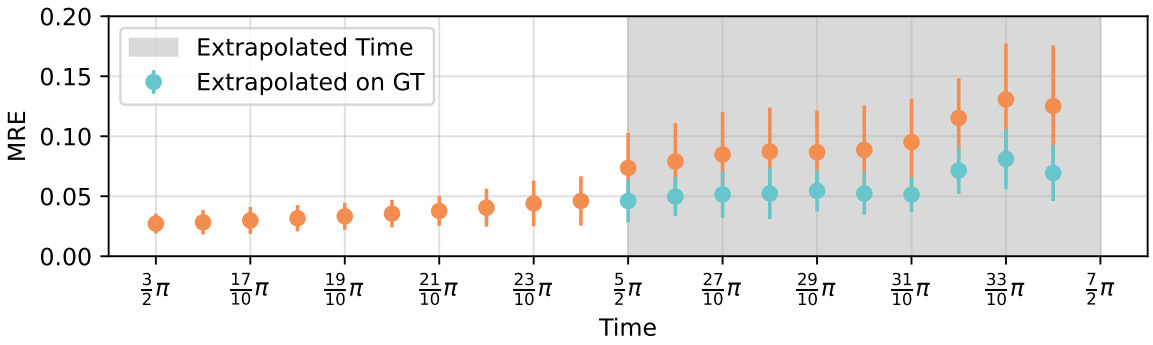



Figure 5: Mean Relative Error (MRE) for predicted Hamiltonian observables in an 8-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. For extrapolated time predictions on ground truth (GT), the MRE is 6.46%. This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 11.18% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $3.096 \times 10^{-6} \pm 2.497 \times 10^{-6}$ and for future time predictions over the same interval on ground truth as $4.212 \times 10^{-6} \pm 3.333 \times 10^{-6}$.

New Figure 4:

**(a)**




**(b)**

| **n** | MRE($24\pi/10$) | MRE − GT($5\pi/2$) | MRE − GT($31\pi/10$) | MRE − GT($34\pi/10$) |
|---|---|---|---|---|
| 8 | $0.0675 \pm 0.0305$ | $0.0242 \pm 0.0114$ | $0.0815 \pm 0.0484$ | $0.1057 \pm 0.0573$ |
| 20 | $0.0460 \pm 0.0208$ | $0.0459 \pm 0.0176$ | $0.0515 \pm 0.0150$ | $0.0704 \pm 0.0247$ |

Figure 6: **(a)** Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. The MRE is 5.8% over the extrapolated time predictions on ground truth (GT). This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 9.6% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $2.16 \times 10^{-9} \pm 2.15 \times 10^{-9}$ and for future time predictions over the same interval on ground truth as $5.34 \times 10^{-9} \pm 4.13 \times 10^{-9}$. **(b)** The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ for systems with 8 and 20 qubits for hamiltonian observables.

5. In the introduction, under "Our Approach", it is stated that data from "...actual quantum computations..." is employed in the training of these FNOs. Based on descriptions in the main text, and methods, this does not appear to be true. As such, either the source and methods for obtaining such data should be clearly stated, or this statement must be revised.

**Author Comment R1.A5:**
We thank the referee for pointing this out. The statement in the Introduction has been revised to accurately reflect our data-generation process. We now clarify that the FNOs are trained on temporal data obtained from classical simulations of quantum dynamics (not from actual quantum computations). The corresponding text in the manuscript has been updated accordingly.

Original discussion:
*We employ temporal data from actual quantum computations to train FNOs, enabling them to capture system dynamics, extrapolate future time intervals, and generate predictions for analogous inputs.*

New discussion:
*We employ temporal data generated from high-fidelity classical simulations of quantum dynamics to train FNOs, enabling them to capture system evolution, extrapolate to future time intervals, and generalize for analogous inputs. This classical-data-driven approach ensures that the model learns accurate dynamics in a noise-free setting, while providing a foundation for future extensions that could use data directly from*

*actual quantum computations to extrapolate dynamics beyond the coherence limit.*

(B) Minor Comments:

1. At the end of the introduction, under "Our approach", authors mention "the U-Net", which has yet to be described at this point in the manuscript.

**Author Comment R1.B1:**
We thank the Referee for pointing this out. We have added a description of the U-Net in Section 1, highlighting it as a convolutional encoder–decoder with skip connections that captures multi-scale features while preserving fine spatial details.

Original discussion:
*Due to its discretization-invariance, the FNO maintains exceptional accuracy, with error rates as low as 0.04% on the finer grid, while the U-Net shows a significantly increased error rate of 51.70% on the finer discretization.*

New discussion:
*Due to its discretization-invariance, the FNO maintains exceptional accuracy, achieving error rates as low as 0.04% on the finer grid. For comparison, we include a U-Net baseline, a widely used convolutional encoder–decoder architecture with skip connections cite[williams2023unified]. Due to its fixed-size kernels, U-Net learns patterns at a fixed resolution and does not learn the underlying operator cite[liuschiaffini2024]. When evaluated on the same finer discretization, the U-Net exhibits a substantially higher error rate of 51.70%. This contrast highlights that while U-Net can reproduce patterns at fixed resolution, it does not learn the underlying operator, emphasizing the advantage of FNOs for capturing system dynamics and enabling accurate extrapolation. As shown in Figure 3, the FNO markedly outperforms the U-Net in the extrapolation regime. The U-Net performs poorly, particularly on low-energy wavefunction inputs, with a mean fidelity of* 0.1501*, whereas the FNO captures the underlying structure and extrapolates with high accuracy, achieving a mean fidelity of* 0.8893*.*

2. In Section 1B, the random n-particle wavefunctions are described as being sampled from a uniform distribution; this process is unclear.

**Author Comment R1.B2:**
We thank the Referee for the comment. In Section 1B, we now clarify that each complex amplitude of the random $n$-particle wavefunctions is independently sampled from a uniform distribution on $[0, 1)$, and the wavefunction is subsequently normalized to unit $L^2$ norm.

Original discussion in Section 1B:
*In this approach, the full state space of a quantum wavefunction is used as input, with two distinct types of wavefunctions considered. The first type consists of complex-valued normalized wavefunctions for n particles, randomly generated from a uniform distribution between* $-1$ *and* $1$*.*

New discussion in Section 1B:
*In this approach, the full state space of a quantum wavefunction is used as input, with two distinct types of wavefunctions considered. The first type consists of complex-valued normalized wavefunctions for n particles. Each amplitude is generated by independently sampling its real and imaginary parts from a uniform distribution on the interval* $[0, 1)$ *using* `torch.rand`*. The resulting wavefunction is then normalized*

*to have unit $L^2$ norm.*

3. In Section 1B1, when discussing extrapolation with random inputs, its stated that various time intervals (VTIs) are not necessary for training. While the explanation is intuitive, the VTI results for random inputs are far superior to those without, as seen in Figure 2a; this should be addressed in the manuscript. Furthermore, it is stated in the text that even without VTI, the mean fidelity at extrapolated times is above 94%. It is unclear where this number comes from, seeing as the fidelity in Figure 2 falls below this value. Is it true then only in the case of 4-qubits?

**Author Comment R1.B3:**
We thank the Referee for the comment. The main text has been updated to clarify the role of VTI. For 4-qubit random inputs without VTI, the mean fidelity at extrapolated times is about 97%, and for 8-qubit random inputs it is around 91%. We have also added an adequate discussion on using VTI for random inputs and low energy inputs in Section 1B1.

Original discussion in Section 1B1:
*We test VTI when training models on random inputs. However, random inputs do not need to be trained across multiple intervals, as they already occupy a wide array of energy regions and can thus successfully extrapolate to later times with a fidelity greater than* 94%.

New discussion in Section 1B1:
*When VTI is applied, low-energy inputs achieve substantially higher mean fidelities at extrapolated times, reaching* 0.9992 *for 4-qubit and* 0.9430 *for 8-qubit states. We also test VTI when training models on random inputs. Random inputs do not strictly require multiple intervals, as they already occupy a wide array of energy regions and can successfully extrapolate to later times. For example, 4-qubit random inputs without VTI reach a mean fidelity of about* 97%*, while 8-qubit random inputs achieve around* 91%*. Nonetheless, applying VTI even for random inputs further improves performance, achieving mean fidelities as high as* 99.99% *for 8 qubits, highlighting that multi-interval training can still be beneficial.*
*FNO performs better on random wavefunctions as these states already span a wide range of energy modes and configurations. Each random state effectively provides the model with information across the entire Hilbert space, allowing it to learn the dynamics for all relevant energy regions in a single step. In contrast, low-energy states are initially localized to only a small subset of modes, and accurately predicting their future evolution requires the model to capture how population spreads to higher-energy modes over time. This highlights why VTI is necessary for accurately extrapolating low-energy states, whereas random states achieve satisfactory results even when the model is trained on a single time step.*

4. The embedding in Section 1B2 is described as using "... the values of two sinusoidal functions applied to the time-grid...". More detail on what these sinusoidal functions are, and how they're chosen, should be provided.

**Author Comment R1.B4:**
We thank the referee for the suggestion. The positional embedding has been clarified to explicitly describe the use of a single cosine–sine pair over a discretized time grid and added in Section 1B2.

Original discussion in Section 1B2:
*For the positional embedding* $\mathrm{Embed}_{\mathrm{time}}$*, we use the values of two sinusoidal functions applied to the time-grid in order to assist the FNO in learning temporal patterns.*

New discussion in Section 1B2:
*For the positional embedding* $\mathrm{Embed}_{\mathrm{time}}$*, we use a sinusoidal encoding to facilitate the FNO in learning temporal patterns. Given a discretized time grid with* $m$ *equidistant points, indexed by* $l \in \{0, \ldots, m-1\}$*, the positional embedding is constructed using a single cosine–sine pair. With two embedding channels and a fixed frequency* $\omega = 1$*, each time index* $l$ *is mapped as*

$$\mathrm{Embed}_{\mathrm{time}}(l) = \big[\cos(\omega l),\ \sin(\omega l)\big]. \tag{1}$$

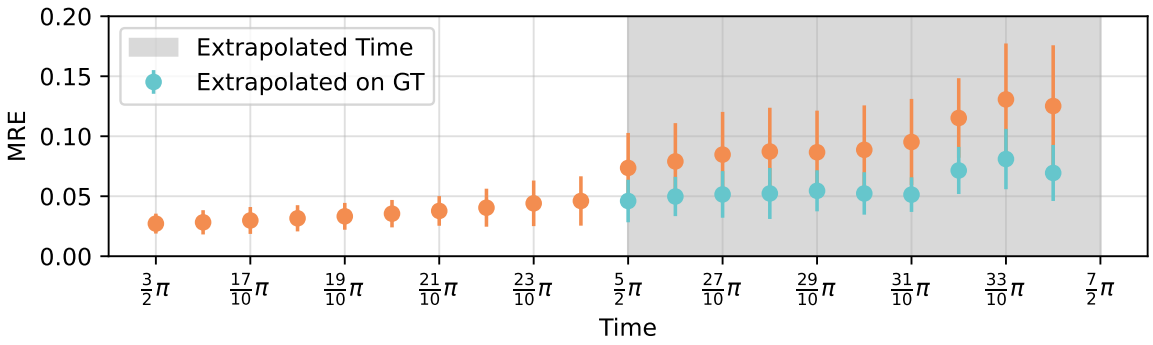
*This results in a positional embedding tensor of shape* $(1, 2, m)$*, where the two channels correspond to the cosine and sine components evaluated at each time step. By mapping time onto a smooth periodic representation, the embedding allows the model to capture the relative ordering of temporal points while preserving continuity across the discretized interval cite[li2020fourier,FNOPracticalReview]. Although the approach can be generalized to multiple frequencies by increasing the number of channels, in this work we use the minimal two-channel formulation.*

5. There appears to be a typo in the caption to Figure 4, where the extrapolated interval is given as "[2/2 $\pi$, 7/2 $\pi$]".

**Author Comment R1.B5:**
We thank the referee for pointing out this typo. The caption of Figure 4 has been corrected, and the extrapolated interval is now given as $[\frac{5}{2}\pi, \frac{7}{2}\pi]$.

New Figure 4:

**(a)**




**(b)**

| **n** | MRE($24\pi/10$) | MRE − GT($5\pi/2$) | MRE − GT($31\pi/10$) | MRE − GT($34\pi/10$) |
|---|---|---|---|---|
| 8 | $0.0675 \pm 0.0305$ | $0.0242 \pm 0.0114$ | $0.0815 \pm 0.0484$ | $0.1057 \pm 0.0573$ |
| 20 | $0.0460 \pm 0.0208$ | $0.0459 \pm 0.0176$ | $0.0515 \pm 0.0150$ | $0.0704 \pm 0.0247$ |

Figure 7: **(a)** Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. The MRE is 5.8% over the extrapolated time predictions on ground truth (GT). This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 9.6% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $2.16 \times 10^{-9} \pm 2.15 \times 10^{-9}$ and for future time predictions over the same interval on ground truth as $5.34 \times 10^{-9} \pm 4.13 \times 10^{-9}$. **(b)** The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ for systems with 8 and 20 qubits for hamiltonian observables.

---

Report of the Second Referee – COMMSPHYS-24-1031

---

(0) The paper concerns simulating dynamics of quantum systems using Machine Learning (ML) techniques. In particular, the authors focus in 1D spin system and use the tool of Fourier Neural Operators (FNOs).

While the topic is timely, and ML-derived techniques either are or could be of potentially impact, the manuscript has a number of serious flaws in regards to methodology, execution and presentation, which in my opinion at present should preclude publication. My main concerns are regarding the actual validity of the method for the larger systems, the confusing presentation, and the complete lack of any attempt at a theoretical understanding of the method itself.

**Author Comment R2.0:**

We thank the Referee for their careful reading of our manuscript and for highlighting important aspects regarding methodology, presentation, and the theoretical understanding of our approach. We appreciate the constructive feedback and have thoroughly revised the manuscript to address these concerns. In particular, we have clarified the methodology, improved the presentation of figures and results, provided additional details on the implementation and scalability of our approach.

Specifically:
1. The system sizes considered are tiny: 4 and 8 qubits. This is important for two reasons:
(A) first with regards to claims and motivations, which prominently mention difficulty in simulating larger systems — at present I don't find any results in the manuscript which give credence to the idea that FNOs allow to address larger quantum systems.

**Author Comment R2.1A:**
We have now included results for both 8-qubit and 20-qubit systems in Figure 4 using Hamiltonian observables as inputs, which is a more practical and scalable approach as compared to using all $2^n$ amplitudes as inputs. The larger 20-qubit system demonstrates that the FNO can scale to substantially larger quantum many-body systems, addressing the Referee's concern about the method being limited to small sizes. The new discussion emphasizes that the FNO maintains accurate extrapolation performance as system size increases. Specifically, we report relative errors of 6.46% for 8 qubits and 5.8% for 20 qubits for predictions beyond the training time interval. Furthermore, the increase in training data is modest (from 18,000 to 22,500 samples), demonstrating the viability of our approach despite the exponential growth of the Hilbert space. We have also added explicit details in Section 1C describing how the time-evolved expectation-value data is generated for the 20-qubit system, providing strong evidence of the method's scalability and performance across larger system sizes.

New discussion in Section 1C:
*For our experiments, we consider system sizes of 8 and 20 qubits. For the larger system (20 qubits), the time-evolved expectation-value data is generated using the `CUDA-Q` library with Trotterization, since exact simulation at these scales is computationally intractable.*

Old Figure 4:

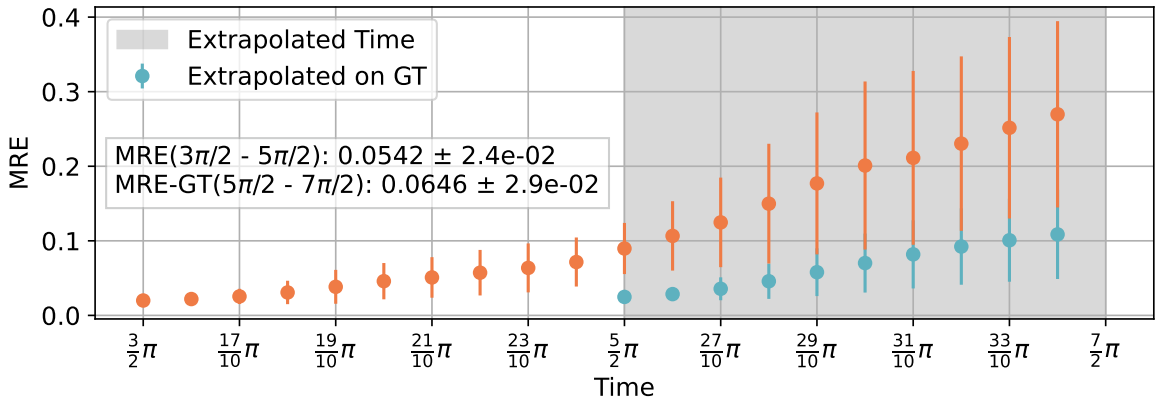
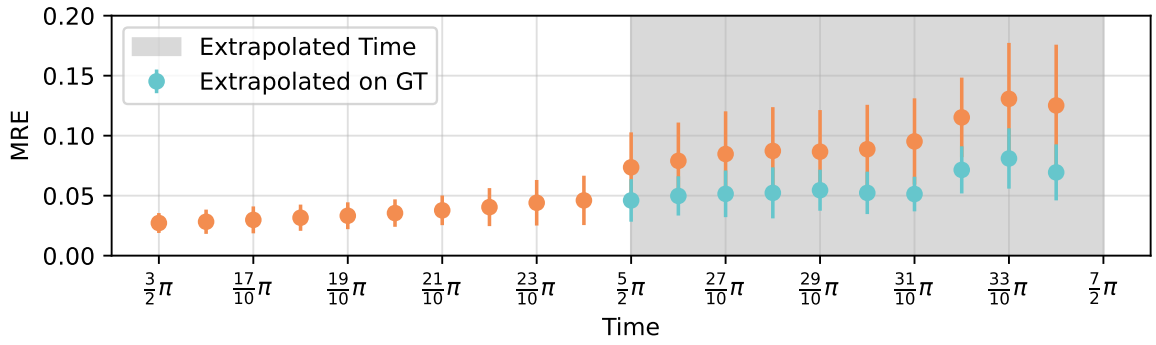



Figure 8: Mean Relative Error (MRE) for predicted Hamiltonian observables in an 8-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. For extrapolated time predictions on ground truth (GT), the MRE is 6.46%. This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 11.18% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $3.096 \times 10^{-6} \pm 2.497 \times 10^{-6}$ and for future time predictions over the same interval on ground truth as $4.212 \times 10^{-6} \pm 3.333 \times 10^{-6}$.

New Figure 4:

**(a)**




**(b)**

| **n** | MRE($24\pi/10$) | MRE − GT($5\pi/2$) | MRE − GT($31\pi/10$) | MRE − GT($34\pi/10$) |
|---|---|---|---|---|
| 8 | 0.0675 ± 0.0305 | 0.0242 ± 0.0114 | 0.0815 ± 0.0484 | 0.1057 ± 0.0573 |
| 20 | 0.0460 ± 0.0208 | 0.0459 ± 0.0176 | 0.0515 ± 0.0150 | 0.0704 ± 0.0247 |

Figure 9: **(a)** Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. The MRE is 5.8% over the extrapolated time predictions on ground truth (GT). This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 9.6% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $2.16 \times 10^{-9} \pm 2.15 \times 10^{-9}$ and for future time predictions over the same interval on ground truth as $5.34 \times 10^{-9} \pm 4.13 \times 10^{-9}$. **(b)** The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ for systems with 8 and 20 qubits for hamiltonian observables.

Original discussion in Section 1C:

*As seen in Figure 4, with a system of 8 qubits we predict future observables with a relative error of* 6.46%, *even when extending the prediction to double the train time horizon on ground truth.*

New discussion in Section 1C:

*As seen in Figure 4, the FNO maintains consistent and accurate performance even as the system size increases substantially. For 8- and 20-qubit systems, we predict future observables with relative errors of* 6.46% *and* 5.8%, *respectively, even when extending the prediction to twice the training time horizon on ground truth data. The model exhibits no degradation in accuracy with increasing system size, and its extrapolation performance remains stable across all settings. This strongly demonstrates that FNOs can scale effectively to larger quantum many-body systems while preserving error rates in both the training regime and long-time extrapolation. It is also noteworthy that the required volume of training data does not scale in proportion to the exponential growth of the Hilbert space. For instance, we use 18,000 training samples for the 8-qubit system and only 22,500 for the 20-qubit system. This modest increase, despite the dramatic growth in system dimensionality, underscores the data efficiency of the FNO architecture and further illustrates its practical scalability for large-scale quantum dynamical prediction.*

(B) Second, with such small system sizes the dimensionality of the Hilbert space in which the wave function lives is no larger than 256 — comparable in size to the dimensionality of the parameter space of the ML models the authors use (if I'm not mistaken the Authors use e.g. 128 Fourier modes), so the fundamental issue of the "curse of dimensionality" for quantum system is not even really encountered in the numerical experiments in the paper. It is critical to perform larger scale experiments, and in particular to verify the scaling of the fidelity with the increase of the number of the qubits. I notice that the 8-qubit results (still a minute system well within reach of ED methods) are already substantially worse than those for 4 qubits: for what the Authors call "low energy" inputs the fidelity already drops by an order of magnitude (however it is presented, more on that below). I am not at all convinced, based on the presented results, that the methods would work reasonably well even on the simulatable scales of 20-30 qubits, much less on scales the authors mention as their motivations.

**Author Comment R2.1B:**
We have now included results for a 20-qubit system in Figure 4. Specifically, we conducted large-scale experiments using Hamiltonian observables as inputs, i.e., the expectation values of the relevant operators. This approach is more practical than using full wavefunction inputs, since the input size for the latter grows exponentially with the number of qubits for each training sample. The larger 20-qubit system demonstrates that the FNO can scale to substantially larger quantum many-body systems, addressing the Referee's concern about the method being limited to small sizes. The new discussion emphasizes that the FNO maintains accurate extrapolation performance as system size increases. Specifically, we report relative errors of 6.46% for 8 qubits and 5.8% for 20 qubits for predictions beyond the training time interval. Furthermore, the increase in training data is modest (from 18,000 to 22,500 samples), demonstrating the viability of our approach despite the exponential growth of the Hilbert space. We have also added explicit details in Section 1C describing how the time-evolved expectation-value data is generated for the 20-qubit system, providing strong evidence of the method's scalability and performance across larger system sizes.

New discussion in Section 1C:
*For our experiments, we consider system sizes of 8 and 20 qubits. For the larger system (20 qubits), the time-evolved expectation-value data is generated using the `CUDA-Q` library with Trotterization, since exact simulation at these scales is computationally intractable.*

Old Figure 4:

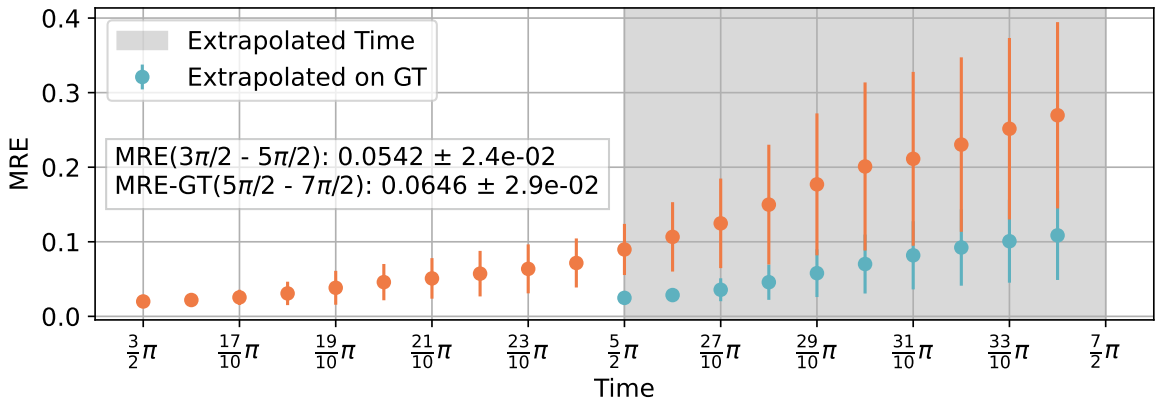
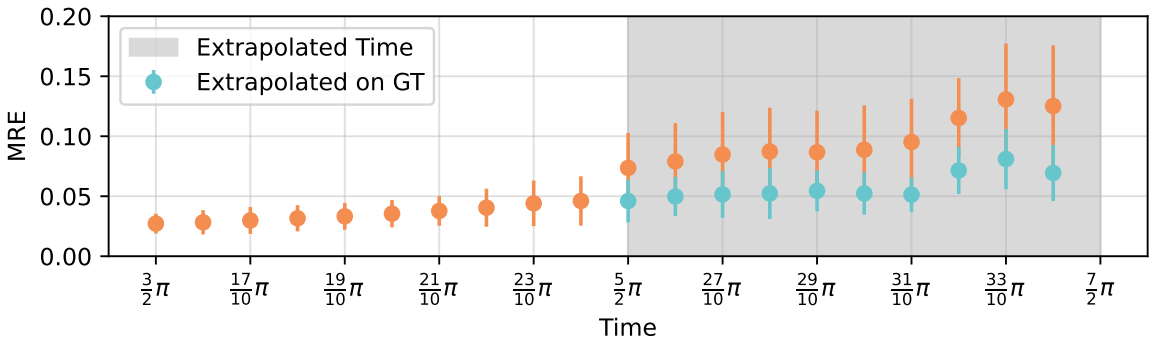



Figure 10: Mean Relative Error (MRE) for predicted Hamiltonian observables in an 8-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. For extrapolated time predictions on ground truth (GT), the MRE is 6.46%. This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 11.18% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $3.096 \times 10^{-6} \pm 2.497 \times 10^{-6}$ and for future time predictions over the same interval on ground truth as $4.212 \times 10^{-6} \pm 3.333 \times 10^{-6}$.

New Figure 4:

**(a)**




**(b)**

| **n** | MRE($24\pi/10$) | MRE − GT($5\pi/2$) | MRE − GT($31\pi/10$) | MRE − GT($34\pi/10$) |
|---|---|---|---|---|
| 8 | $0.0675 \pm 0.0305$ | $0.0242 \pm 0.0114$ | $0.0815 \pm 0.0484$ | $0.1057 \pm 0.0573$ |
| 20 | $0.0460 \pm 0.0208$ | $0.0459 \pm 0.0176$ | $0.0515 \pm 0.0150$ | $0.0704 \pm 0.0247$ |

Figure 11: **(a)** Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. The MRE is 5.8% over the extrapolated time predictions on ground truth (GT). This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 9.6% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $2.16 \times 10^{-9} \pm 2.15 \times 10^{-9}$ and for future time predictions over the same interval on ground truth as $5.34 \times 10^{-9} \pm 4.13 \times 10^{-9}$. **(b)** The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ for systems with 8 and 20 qubits for hamiltonian observables.

Original discussion in Section 1C:
*As seen in Figure 4, with a system of 8 qubits we predict future observables with a relative error of* 6.46%, *even when extending the prediction to double the train time horizon on ground truth.*

New discussion in Section 1C:
*As seen in Figure 4, the FNO maintains consistent and accurate performance even as the system size increases substantially. For 8- and 20-qubit systems, we predict future observables with relative errors of* 6.46% *and* 5.8%, *respectively, even when extending the prediction to twice the training time horizon on ground truth data. The model exhibits no degradation in accuracy with increasing system size, and its extrapolation performance remains stable across all settings. This strongly demonstrates that FNOs can scale effectively to larger quantum many-body systems while preserving error rates in both the training regime and long-time extrapolation. It is also noteworthy that the required volume of training data does not scale in proportion to the exponential growth of the Hilbert space. For instance, we use 18,000 training samples for the 8-qubit system and only 22,500 for the 20-qubit system. This modest increase, despite the dramatic growth in system dimensionality, underscores the data efficiency of the FNO architecture and further illustrates its practical scalability for large-scale quantum dynamical prediction.*

2. The paper does not contain any attempt at theoretical understanding how, when, and why the method should work for the dynamics of quantum systems. It feeds some physical data into an ML architecture,

including also arbitrary choices (e.g. the number of Fourier modes kept), and observes the final fidelity. The comments about the difficulty os simulating the dynamics of "low energy states" vs. random states seem handwaving and unconvincing. In classical dynamical systems (and in quantum too) there is an understanding that ML-like methods attempting to extrapolate dynamics often manage (at least when they actually make sense) to capture e.g. the leading Koopman modes of the dynamical systems. I would encourage the Authors to compare the results to ones obtained using e.g. Dynamical Mode decomposition (DMD). As it stands, even uf the method would numerically work (and in my opinion it doesn't, at scale) there is understanding or estimates for the error the method makes (other than actually comparing to the exact solution), and no reason to believe it would be capturing "the right thing" when such a comparison would be unavailable.

**Author Comment R2.2:**
We thank the reviewer for raising an important point regarding theoretical understanding and comparison with methods such as Dynamical Mode Decomposition (DMD). We would like to clarify why direct application of DMD in our setting is not straightforward, and how the Fourier Neural Operator (FNO) differs fundamentally in terms of generalization.

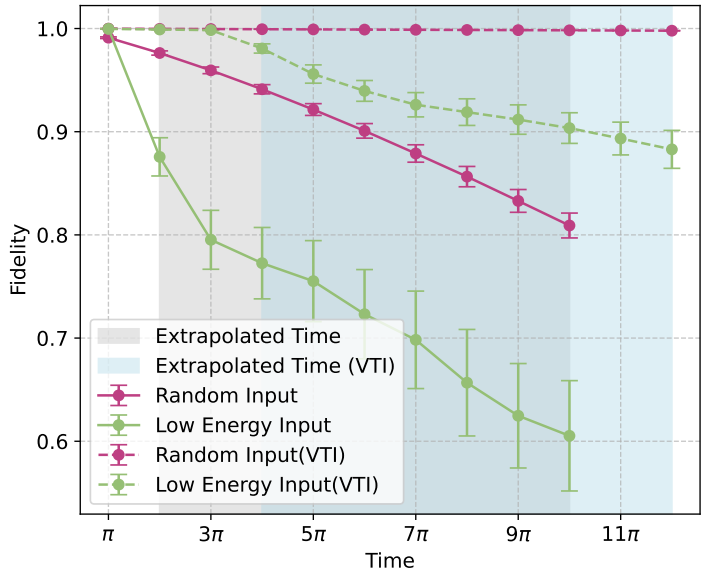
New discussion in Introduction:
*Similarly, a direct comparison with classical trajectory-based methods such as Dynamical Mode Decomposition (DMD) is not feasible. DMD assumes knowledge of the system's initial state and operates on snapshot pairs from a single trajectory, learning a linear operator that describes the evolution of that specific state cite[Zhang2016Evaluating]. In contrast, in our setting, the initial state is unknown, so DMD cannot be directly applied. The FNO, on the other hand, is trained on a diverse set of randomly generated quantum wavefunctions and learns a functional mapping of the time evolution across an entire class of states. This allows FNOs not only to reproduce local features but also to capture the underlying operator governing quantum dynamics, a capability that neither purely pattern-based ML models nor single-trajectory linear methods can provide.*

3. I find the way the data is presented somewhat misleading: we are interested in the simulation fidelity at longer times. Such fidelity always decays in time, essentially monotonically. There is no reason to give in tables the "averaged" fidelity over the whole time interval after training. The only function of this seems to improve the numbers by averaging the infidelity at longer times with the high fidelities at short times, but is otherwise physically meaningless. There are no solid baselines for comparison against any other physical methods given, other than a completely arbitrary, non-standard in physical simulations (and not described in any detail), U-Net machine learning model.

**Author Comment R2.3:**
We thank the referee for this important comment regarding the presentation of the fidelity data. We agree that the physically relevant regime is the long-time behavior, where the fidelity typically decays monotonically, and that simple time-averaged fidelities can obscure this behavior by mixing short- and long-time performance. In the revised manuscript, we therefore remove the tables reporting time-averaged fidelities over the training and extrapolation intervals. Instead, we report fidelities at selected representative time steps within these intervals. We have also added a more detailed discussion of the U-Net baseline in the main text, clarifying both its architecture and our motivation for including it as a benchmark.

Old Figure 2:

**(a)**




**(b)**

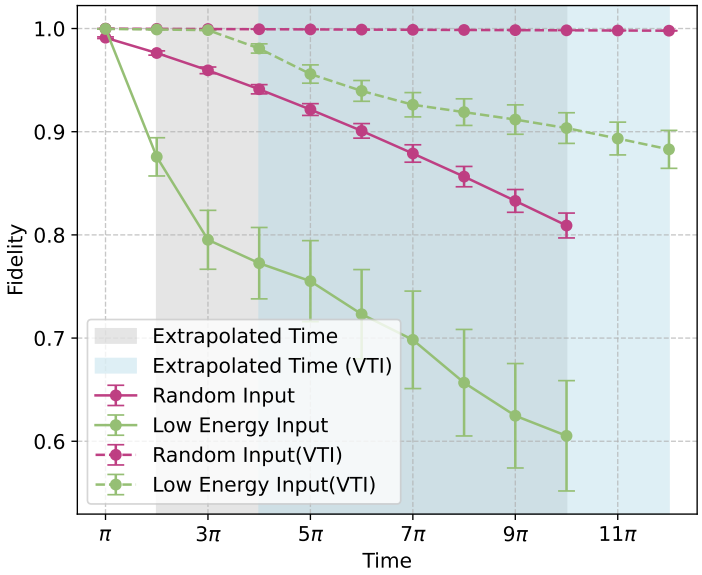
| n | Input | **Fidelity**$[0, \pi]$ | **Fidelity**$[2\pi, 10\pi]$ |
|---|---|---|---|
| 4 | Random | 0.9983 ± 0.0 | 0.9732 ± 2.3e-02 |
| 8 | Random | 0.9920 ± 0.0 | 0.9062 ± 5.8e-02 |

| n | Input | **Fidelity**$[0, 3\pi]$ | **Fidelity**$[4\pi, 12\pi]$ |
|---|---|---|---|
| 4 | Low Energy (VTI) | 0.9999 ± 0.0 | 0.9992 ± 1.0e-03 |
| 8 | Low Energy (VTI) | 0.9996 ± 0.0 | 0.9430 ± 4.2e-02 |

Figure 12: **(a)** Prediction of temporal dynamics in an 8-qubit Heisenberg spin system using the energy-domain architecture in Section 1B1. At time $T = \pi$, the predictions made by the FNO are on unseen data but still within the time range it was trained on. The extrapolated time refers to future time predictions beyond the range on which the FNO was trained. VTI (Various Time Interval) indicates training on multiple time intervals, i.e., $[0, T]$, $[T, 2T]$, and $[2T, 3T]$, instead of training only on the first interval. **(b)** The results show the mean fidelity for time intervals the FNO has seen during training and extrapolated time intervals for systems with 4 and 8 qubits.

New Figure 2:

**(a)**




**(b)**

**Random initial states**

| **n** | $F(\pi)^{\text{train}}$ | $F(2\pi)^{\text{ext}}$ | $F(6\pi)^{\text{ext}}$ | $F(10\pi)^{\text{ext}}$ |
|---|---|---|---|---|
| 4 | $0.9997 \pm 0.0001$ | $0.9991 \pm 0.0003$ | $0.9949 \pm 0.0026$ | $0.9894 \pm 0.0060$ |
| 8 | $0.9911 \pm 0.0009$ | $0.9762 \pm 0.0023$ | $0.9010 \pm 0.0074$ | $0.8101 \pm 0.0129$ |

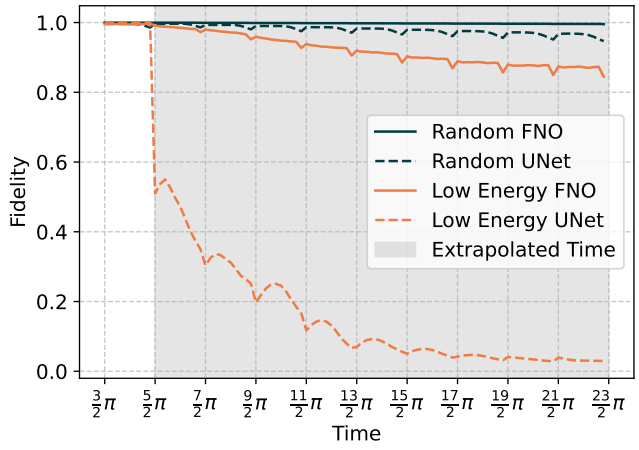
**Low-energy states (VTI)**

| **n** | $F(3\pi)^{\text{train}}$ | $F(4\pi)^{\text{ext}}$ | $F(8\pi)^{\text{ext}}$ | $F(12\pi)^{\text{ext}}$ |
|---|---|---|---|---|
| 4 | $0.9999 \pm 0.0001$ | $0.9999 \pm 0.0001$ | $0.9994 \pm 0.0005$ | $0.9984 \pm 0.0014$ |
| 8 | $0.9984 \pm 0.0002$ | $0.9806 \pm 0.0047$ | $0.9202 \pm 0.0125$ | $0.8836 \pm 0.0184$ |

Figure 13: **(a)** Prediction of temporal dynamics in an 8-qubit Heisenberg spin system using the energy-domain architecture in Section 1B1. At time $T = \pi$, the predictions made by the FNO are on unseen data but still within the time range it was trained on. The extrapolated time refers to future time predictions beyond the range on which the FNO was trained. VTI (Various Time Interval) indicates training on multiple time intervals, i.e., $[0, T]$, $[T, 2T]$, and $[2T, 3T]$, instead of training only on the first interval. **(b)** The results show the fidelity at specific time steps for 4 and 8 qubits, including times within the training range as well as extrapolated future times beyond the training interval. Superscripts indicate whether the particular time step is within the training range (train) or an extrapolated time (ext).

Old Figure 3:

**(a)**




**(b)** FNO

| n | Input Type | Fidelity$[\frac{3}{2}\pi, \frac{5}{2}\pi]$ | Fidelity$[\frac{5}{2}\pi, \frac{23}{2}\pi]$ |
|---|---|---|---|
| 4 | Random | 0.9999 ± 0.0 | 0.9996 ± 3.00e-03 |
| 4 | Low Energy | 0.9999 ± 0.0 | 0.9992 ± 1.00e-03 |
| 8 | Random | 0.9996 ± 2.43e-05 | 0.9978 ± 7.81e-05 |
| 8 | Low Energy | 0.9997 ± 3.83e-05 | 0.9179 ± 9.55e-03 |

**(c)** U-Net

| n | Input Type | Fidelity$[\frac{3}{2}\pi, \frac{5}{2}\pi]$ | Fidelity$[\frac{5}{2}\pi, \frac{23}{2}\pi]$ |
|---|---|---|---|
| 4 | Random | 0.9966 ± 6.52e-05 | 0.9580 ± 8.00e-03 |
| 4 | Low Energy | 0.9999 ± 0.0 | 0.7876 ± 6.10e-02 |
| 8 | Random | 0.9945 ± 5.37e-05 | 0.9790 ± 8.42e-05 |
| 8 | Low Energy | 0.9991 ± 3.59e-05 | 0.1492 ± 6.50e-03 |

Figure 14: **(a)** Prediction of temporal dynamics in an 8-qubit Heisenberg spin system using the time-domain architecture, with FNO and U-Net. The input time interval is denoted as $[0, \frac{3}{2}\pi]$, as illustrated in Figure 1 d), while the extrapolated time refers to future time predictions beyond the trained range. **(b)** The results present the mean fidelity for both the training time (i.e., the time intervals the FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{23}{2}\pi]$ for systems with 4 and 8 qubits, across different input types. **(c)** The results show the mean fidelity using U-Net for similar inputs and time intervals.

New Figure 3:

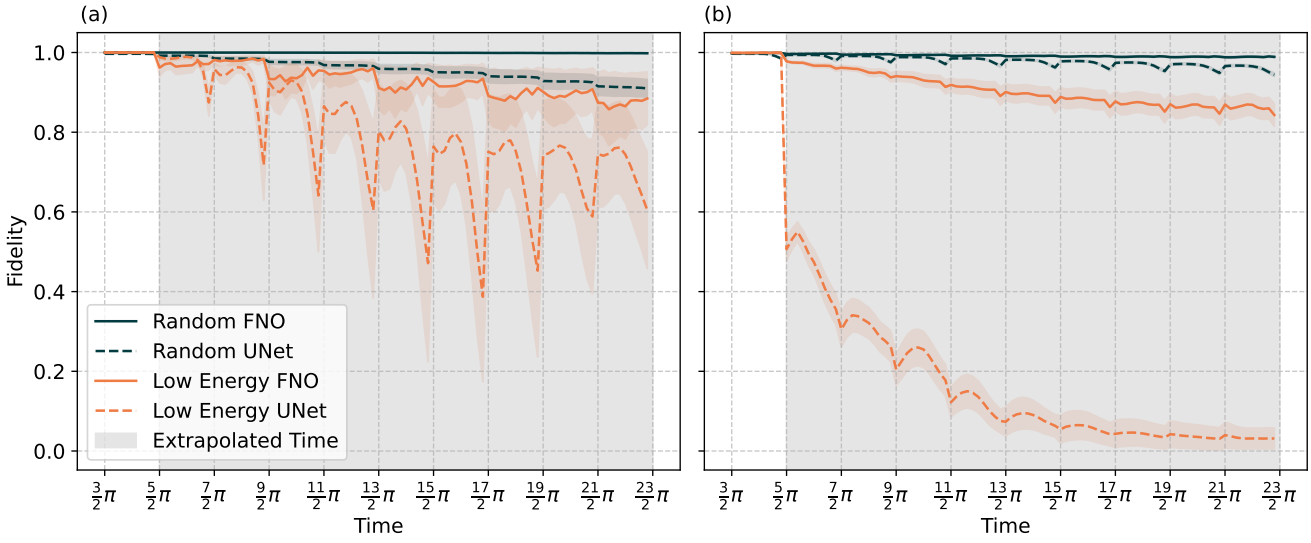



Figure 15: **(a)** Prediction of temporal dynamics in a four-qubit Heisenberg spin system using the time-domain architecture in Section 1B2. The model is trained on the time interval $[0, 3\pi/2]$ as illustrated in Figure 1d, while the extrapolated time refers to future predictions beyond the trained range. Performance for both the random and low-energy wavefunction initial states using the FNO is benchmarked against a U-Net baseline. The shaded regions around each curve indicate one standard deviation. The FNO accurately captures the underlying operator and extrapolates reliably to later times, while the U-Net exhibits poor extrapolation, especially for low-energy input states. **(b)** Corresponding results for an eight-qubit Heisenberg spin system.

Old Figure 4:

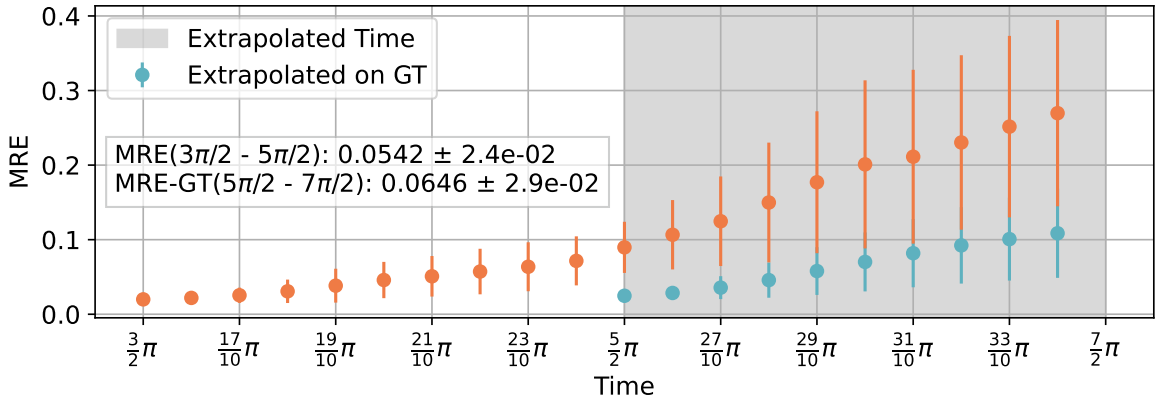
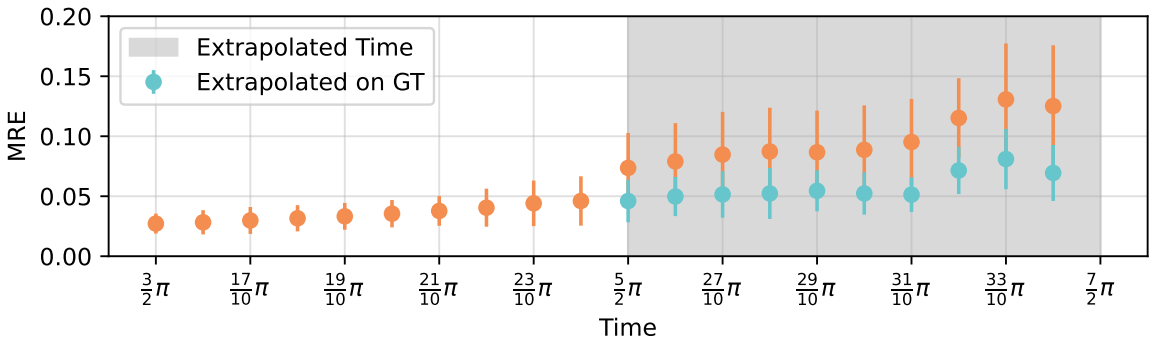



Figure 16: Mean Relative Error (MRE) for predicted Hamiltonian observables in an 8-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. For extrapolated time predictions on ground truth (GT), the MRE is 6.46%. This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 11.18% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $3.096 \times 10^{-6} \pm 2.497 \times 10^{-6}$ and for future time predictions over the same interval on ground truth as $4.212 \times 10^{-6} \pm 3.333 \times 10^{-6}$.

New Figure 4:

**(a)**




**(b)**

| **n** | MRE($24\pi/10$) | MRE − GT($5\pi/2$) | MRE − GT($31\pi/10$) | MRE − GT($34\pi/10$) |
|---|---|---|---|---|
| 8 | $0.0675 \pm 0.0305$ | $0.0242 \pm 0.0114$ | $0.0815 \pm 0.0484$ | $0.1057 \pm 0.0573$ |
| 20 | $0.0460 \pm 0.0208$ | $0.0459 \pm 0.0176$ | $0.0515 \pm 0.0150$ | $0.0704 \pm 0.0247$ |

Figure 17: **(a)** Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1 d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. The MRE is 5.8% over the extrapolated time predictions on ground truth (GT). This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 9.6% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $2.16 \times 10^{-9} \pm 2.15 \times 10^{-9}$ and for future time predictions over the same interval on ground truth as $5.34 \times 10^{-9} \pm 4.13 \times 10^{-9}$. **(b)** The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ for systems with 8 and 20 qubits for hamiltonian observables.

Original discussion:
*Due to its discretization-invariance, the FNO maintains exceptional accuracy, with error rates as low as 0.04% on the finer grid, while the U-Net shows a significantly increased error rate of 51.70% on the finer discretization.*

New discussion:
*Due to its discretization-invariance, the FNO maintains exceptional accuracy, achieving error rates as low as 0.04% on the finer grid. For comparison, we include a U-Net baseline, a widely used convolutional encoder–decoder architecture with skip connections cite[williams2023unified]. Due to its fixed-size kernels, U-Net learns patterns at a fixed resolution and does not learn the underlying operator cite[liuschiaffini2024]. When evaluated on the same finer discretization, the U-Net exhibits a substantially higher error rate of 51.70%. This contrast highlights that while U-Net can reproduce patterns at fixed resolution, it does not learn the underlying operator, emphasizing the advantage of FNOs for capturing system dynamics and enabling accurate extrapolation. As shown in Figure 3, the FNO markedly outperforms the U-Net in the extrapolation regime. The U-Net performs poorly, particularly on low-energy wavefunction inputs, with a mean fidelity of* 0.1501*, whereas the FNO captures the underlying structure and extrapolates with high accuracy, achieving a mean fidelity of* 0.8893*.*

4. The explanation of the technical parts are unclear and the notation is confusing and in parts likely

incorrect.
(A) For instance: Eq. (5) states that the object "I" is a complex vector of dimension $C^{2*2^n}$, and that it contains some object $S_0$ "which corresponds to the basis states ($S$)", and then lists a vector of $2^n$ kets (in Dirac's notation), each of which is presumably a computational basis state, itself of dimension $2^n$. This doesn't seem to make sense, even from the point of dimensionality — I assume that the notation is just incorrect and what the Authors really mean is that $S_0$ is vector of $2^n$ amplitudes in the computational basis and not of the vectors itself.

**Author Comment R2.4A:**
We thank the referee for pointing out this important detail. The notation and explanation of $S_0$ have been clarified to avoid confusion. Specifically, $S_0$ now explicitly denotes the vector of complex amplitudes of the wavefunction in the computational basis, rather than a list of kets. The main text has been updated accordingly to reflect this corrected notation, ensuring that the dimensionality and meaning of the FNO input **I** are now clear.

Original discussion in Section 1B1:
*The term $S_0$ corresponds to the basis states $S$ of a $2^n$ wavefunction, $[|\psi_1\rangle, |\psi_2\rangle, \cdots, |\psi_{2^n}\rangle]$, at the initial time $t = 0$.*

New discussion in Section 1B1:
*Here, $S_0$ is the vector of complex amplitudes of the $2^n$ wavefunction in the computational basis at initial time $t = 0$, $S_0 = [c_1, c_2, \ldots, c_{2^n}]$, with $\sum_{i=1}^{2^n} |c_i|^2 = 1$.*

(B) Further down in Eq. 7 $\psi$ is used in the Fourier transform formula and referred to as the "state" (when probably the amplitude is meant), without using Dirac's notation, and in the "methods" section the basis vectors or amplitudes are denoted by "$\phi_k$" instead.

**Author Comment R2.4B:**
We thank the referee for highlighting this notational inconsistency. The discussion and notation have now been clarified and made consistent with the Methods section. Specifically, Eq. (7) now refers to the Fourier transform of the complex amplitudes $c_k$ of the k-th computational basis state of the wavefunction. This ensures that the description is consistent with the notation used for basis states ($\phi_i$) and amplitudes ($c_i$) in the Methods section.

Original discussion in Section 1B1:
*The Fourier transform on the basis states $S$ of the wavefunction is given by*

$$\frac{1}{2^n} \sum_{k=0}^{2^n-1} \psi_k e^{\frac{-ik\tau}{\hbar}}, \tag{2}$$

*Here, $\psi_k$ is the basis state, $\tau$ can be understood as a time-like variable, and $\hbar$ is the reduced Planck constant.*

New discussion in Section 1B1:
*The Fourier transform on the amplitudes $c_k$ of the wavefunction in the computational basis is given by*

$$\frac{1}{2^n} \sum_{k=0}^{2^n-1} c_k e^{\frac{-ik\tau}{\hbar}}, \tag{3}$$

*where $c_k$ is the complex amplitude of the k-th computational basis state, $\tau$ can be understood as a time-like variable, and $\hbar$ is the reduced Planck constant.*

5. Corresponding to the above, the figure 1 also insufficiently reflects the notations in the text (no subscripts of any sort, which do appear in the formulas), and contains parts which seem incorrect (in fig. 1c the vector of kets given as inputs, which manifestly does not have a dimension given in Eq.5, and does not seem to make sense). Fig.1a contains an image which is also highly confusing with some 2D/3D like picture to illustrate a Heisenberg chain in 1D mixing the $J_x, J_y, J_z$ couplings for the same nearest neighbours along the chains with spatial directions. This is extremely puzzling, to say the least.

**Author Comment R2.5:**
Fig. 1 has been revised to address the raised concerns. In Fig. 1c, the input vector of kets has been corrected to match the dimensions specified in Eq. 5 and Fig. 1a has been replaced with a clearer schematic that accurately represents the 1D Heisenberg chain with nearest-neighbor interactions along the chain. We believe these changes clarify the figure and its connection to the text.

Old Figure 1:

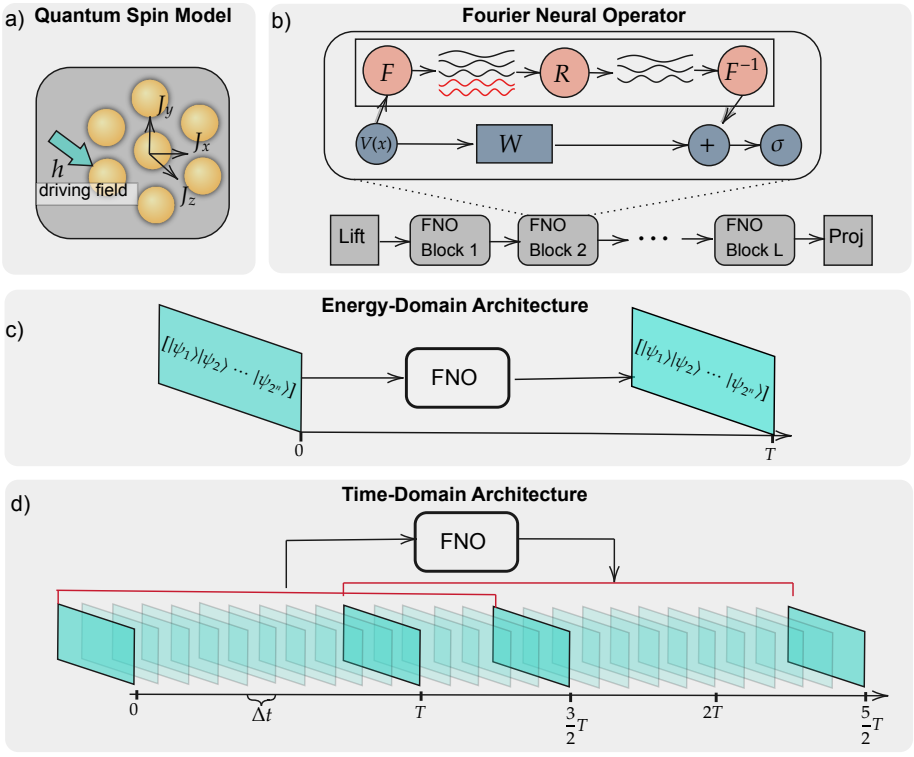



Figure 18: **(a)** Schematic representation of quantum spin system used to model its dynamics using FNOs, where $J_x$, $J_y$, and $J_z$ represent the coupling constants and $h$ denotes the external driving field. **(b)** Diagram illustrating the FNO framework, which consists of multiple FNO blocks performing spectral convolution over the input in the latent space. **(c)** We investigate two distinct architectures for learning dynamics in quantum spin systems. The first is the energy-domain architecture (see Section 1B1), where the input is the wavefunction at the initial time $t = 0$, and the output is the wavefunction evolved to time $t = T$. **(d)** The second is the time-domain architecture (see Section 1B2), which takes the wavefunction evolved over the initial time interval $[0, \frac{3}{2}T]$ (discretized on a grid with width $\Delta t$) as input and produces an output wavefunction over the time interval $[T, \frac{5}{2}T]$.

New Figure 1:

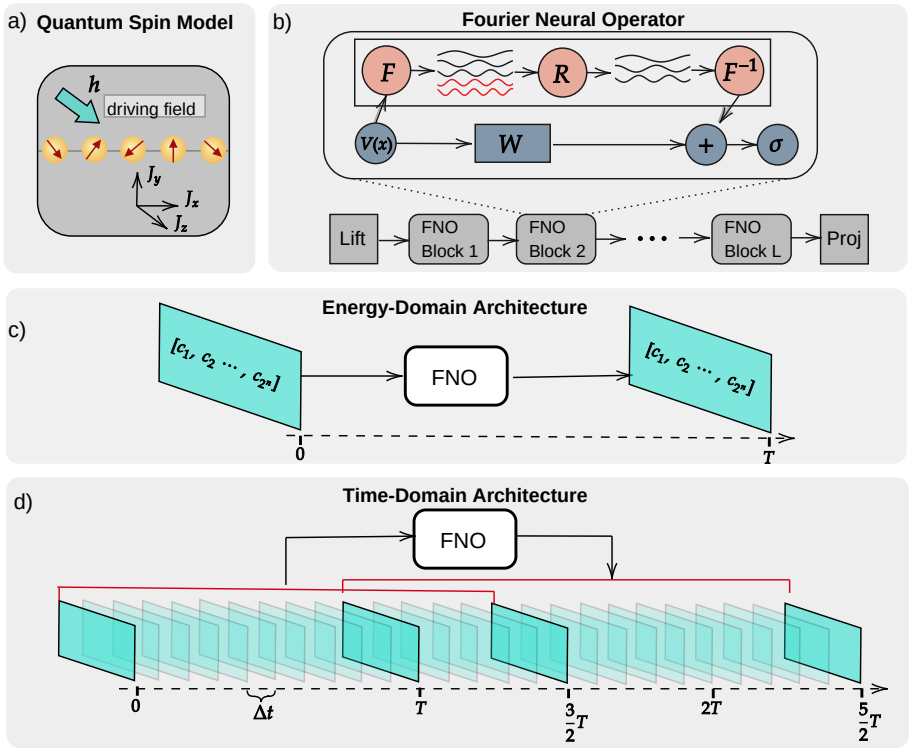



Figure 19: **(a)** Schematic representation of quantum spin system used to model its dynamics using FNOs, where $J_x$, $J_y$, and $J_z$ represent the coupling constants and $h$ denotes the external driving field. **(b)** Diagram illustrating the FNO framework, which consists of multiple FNO blocks performing spectral convolution over the input in the latent space. **(c)** We investigate two distinct architectures for learning dynamics in quantum spin systems. The first is the energy-domain architecture (see Section 1B1), where the input is the wavefunction at the initial time $t = 0$, and the output is the wavefunction evolved to time $t = T$. **(d)** The second is the time-domain architecture (see Section 1B2), which takes the wavefunction evolved over the initial time interval $[0, \frac{3}{2}T]$ (discretized on a grid with width $\Delta t$) as input and produces an output wavefunction over the time interval $[T, \frac{5}{2}T]$.

6. The explanation of the "Hamiltonian Observables" section I.C is insufficient. First, the Authors continuously confuse "observables" with "expectation values of observables". Second, in contradiction to their own statements in the previous section and Fig.2 now they claim that it is the random states which are more difficult to simulate, and they don't consider the "low energy states " anymore. This seems arbitrary and unjustified. The observable method is supposed to allow to tackle even larger systems, but they still confine themselves to 4 and 8 qubits. They do not describe the procedure in any detail, and the "methods" section is of not much help either, additionally containing some arbitrary cutoffs.

**Author Comment R2.6:**
In the revised manuscript, we distinguish explicitly between observables (the Hermitian operators appearing in the Hamiltonian) and the expectation values of these observables, which constitute the actual inputs to the FNO. We have rewritten Section 1C to avoid this ambiguity and to clearly indicate that the model processes expectation values of Hamiltonian terms, not the operators themselves.
We thank the referee for pointing out the typo in Section 1C where we claim that randomly generated states are general and as a result challenging. The generality and challenge pertain to the random Hamiltonian coefficients ($J_z$, $h$), not the wavefunction states. The manuscript has been updated to clarify this point.
We now present results for both 8-qubit and 20-qubit systems (Figure 4). The larger 20-qubit system

demonstrates that the FNO can scale to substantially larger quantum many-body systems, addressing the original concern about the method being limited to small sizes. The new discussion emphasizes that the FNO maintains accurate extrapolation performance as system size increases. Specifically, we report relative errors of 6.46% for 8 qubits and 5.8% for 20 qubits for predictions beyond the training time interval. Furthermore, the increase in training data is modest (from 18,000 to 22,500 samples), demonstrating the viability of our approach despite the exponential growth of the Hilbert space. We have also added explicit details in Section 1C describing how the time-evolved expectation-value data is generated for the 20-qubit system, providing strong evidence of the method's scalability and performance across larger system sizes.

Original discussion in Section 1C:
*Hamiltonian observables refer to the individual terms (observables) from the time-evolution generator (Hamiltonian) of the quantum wavefunction. These observables are obtained using the wavefunction by calculating the expectation values for the individual observable terms. For example, in the Ising model, the observables could include terms like $\sigma_z\sigma_z$ or $\sigma_z$ while in the Heisenberg model, they could include $\sigma_a\sigma_a$ (where $a \in \{x, y, z\}$) and $\sigma_z$. Using observables from the Hamiltonian, instead of the full wavefunction, as inputs and outputs for the FNO greatly enhances its scalability, as there are approximately poly$(n)$ terms in the former and $2^n$ terms in the latter.*

New discussion in Section 1C:
*Hamiltonian observables refer to the individual operator terms that appear in the time-evolution generator (the Hamiltonian) of the system. In practice, we do not use these operators themselves as inputs to the model. Rather, we compute their expectation values with respect to the evolving quantum state and use those quantities as the actual inputs. For example, in the Ising model, the relevant operators include terms such as $\sigma_z\sigma_z$ and $\sigma_z$, while in the Heisenberg model they include $\sigma_a\sigma_a$ for $a \in \{x, y, z\}$ and $\sigma_z$. We evaluate the corresponding expectation values $\langle O_j\rangle_t = \langle\psi(t)|O_j|\psi(t)\rangle$ at each time step, which then form the real-valued input and output channels for the FNO. Using expectation values of Hamiltonian, instead of the full wavefunction, as inputs and outputs for the FNO greatly enhances its scalability, as there are approximately poly$(n)$ terms in the former and $2^n$ terms in the latter.*

Original discussion in Section 1C:
*For the observables, we exclusively utilize the time-domain architecture. In this setup, the input consists of predefined observables and their time-evolved intervals. We provide a detailed description of the specific observables used in Section 3. This input structure mirrors the previously described input $\boldsymbol{I}$ in Eq.9, but instead of wavefunction basis states $S$, it includes observable quantities.*

New discussion in Section 1C:
*For the observable-based inputs, we exclusively utilize only the time-domain architecture. In this configuration, the model receives a fixed set of observable expectation values over the chosen time interval. A detailed specification of the observables appears in Section 3. This input structure mirrors the previously described input $\boldsymbol{I}$ in Eq.9, but instead of $2^n$ wavefunction basis states $S$, it includes a substantially smaller set of observable expectation values. We use the same input time interval, $[0, 3T/2]$, with $T = \pi$ and time step $\Delta t = \pi/10$, as described in Section 1B2.*

Original discussion in Section 1C:
*To evaluate the FNO's capabilities, we first considered the Ising Hamiltonian as described in Eq. (2), where $J_z$ and $h$ are randomly generated. Importantly, in this study, we focus on using random wavefunction states to calculate the expectation values of these observables, making the learning and prediction tasks for the FNO maximally general and, as a result, challenging.*

New discussion in Section 1C:
*To evaluate the FNO's capabilities, we first considered the Ising Hamiltonian as described in Eq. (2), where $J_z$ and $h$ are randomly generated. This introduces variability in the Hamiltonian parameters, ensuring that the learning task is general with respect to the Hamiltonian and hence challenging. Importantly, in this study, we focus on using random wavefunction states to calculate the expectation values of these observables*

New discussion in Section 1C:
*For our experiments, we consider system sizes of 8 and 20 qubits. For the larger system (20 qubits), the time-evolved expectation-value data is generated using the `CUDA-Q` library with Trotterization, since exact simulation at these scales is computationally intractable.*

Old Figure 4:

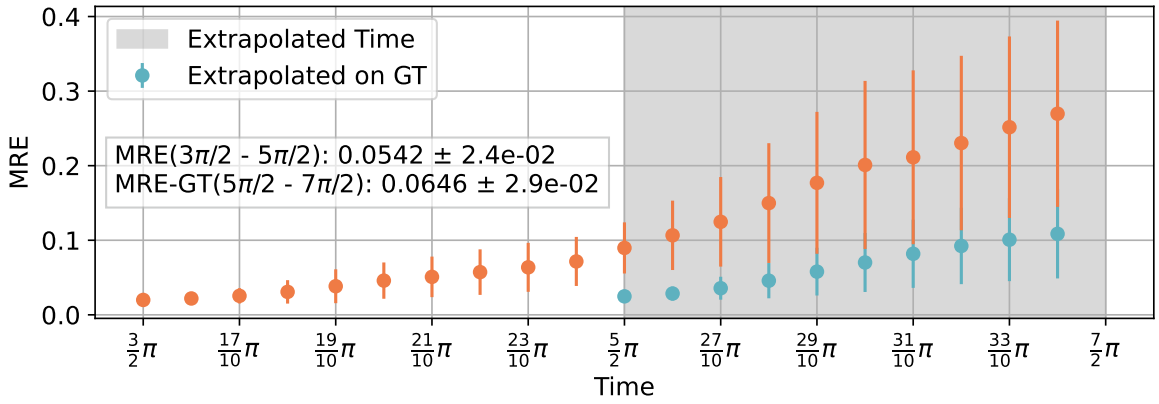
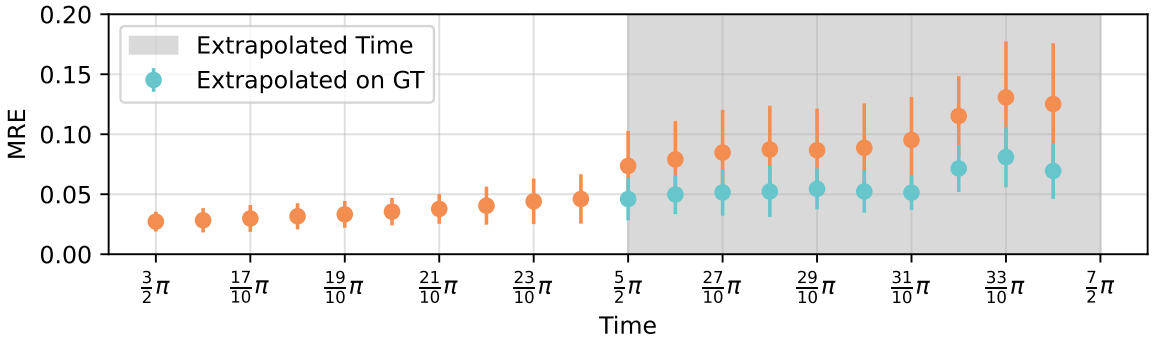



Figure 20: Mean Relative Error (MRE) for predicted Hamiltonian observables in an 8-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. For extrapolated time predictions on ground truth (GT), the MRE is 6.46%. This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 11.18% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $3.096 \times 10^{-6} \pm 2.497 \times 10^{-6}$ and for future time predictions over the same interval on ground truth as $4.212 \times 10^{-6} \pm 3.333 \times 10^{-6}$.

New Figure 4:

**(a)**




**(b)**

| **n** | MRE($24\pi/10$) | MRE − GT($5\pi/2$) | MRE − GT($31\pi/10$) | MRE − GT($34\pi/10$) |
|---|---|---|---|---|
| 8 | $0.0675 \pm 0.0305$ | $0.0242 \pm 0.0114$ | $0.0815 \pm 0.0484$ | $0.1057 \pm 0.0573$ |
| 20 | $0.0460 \pm 0.0208$ | $0.0459 \pm 0.0176$ | $0.0515 \pm 0.0150$ | $0.0704 \pm 0.0247$ |

Figure 21: **(a)** Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. The MRE is 5.8% over the extrapolated time predictions on ground truth (GT). This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 9.6% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $2.16 \times 10^{-9} \pm 2.15 \times 10^{-9}$ and for future time predictions over the same interval on ground truth as $5.34 \times 10^{-9} \pm 4.13 \times 10^{-9}$. **(b)** The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ for systems with 8 and 20 qubits for hamiltonian observables.

Original discussion in Section 1C:
*As seen in Figure 4, with a system of 8 qubits we predict future observables with a relative error of* 6.46%, *even when extending the prediction to double the train time horizon on ground truth.*

New discussion in Section 1C:
*As seen in Figure 4, the FNO maintains consistent and accurate performance even as the system size increases substantially. For 8- and 20-qubit systems, we predict future observables with relative errors of* 6.46% *and* 5.8%, *respectively, even when extending the prediction to twice the training time horizon on ground truth data. The model exhibits no degradation in accuracy with increasing system size, and its extrapolation performance remains stable across all settings. This strongly demonstrates that FNOs can scale effectively to larger quantum many-body systems while preserving error rates in both the training regime and long-time extrapolation. It is also noteworthy that the required volume of training data does not scale in proportion to the exponential growth of the Hilbert space. For instance, we use 18,000 training samples for the 8-qubit system and only 22,500 for the 20-qubit system. This modest increase, despite the dramatic growth in system dimensionality, underscores the data efficiency of the FNO architecture and further illustrates its practical scalability for large-scale quantum dynamical prediction.*

To sum up, despite the motivations and claims mentioning difficulty in simulating larger systems with other methods, such as e.g. tensor networks, I do not find that the Authors convincingly demonstrate that

their results provide an alternative method which is accurate, scales to larger systems and is physically principled. On the contrary, I have very strong doubts about the validity of the method and the quality of the results and their presentation. I find that these are critical faults in the manuscript, and advise a rejection.

---

Report of the Third Referee – COMMSPHYS-24-1031

---

In this work, the authors propose to learn the time evolution of a quantum state using a Fourier Neural Operator. They investigate two architectures, an energy domain one (trained on a single time interval) and time-domain one (trained over multiple time intervals) and show that these architectures can generalize to longer times.

(0) My biggest concern with this work is the scalability and therefore applicability. Although the results seem scientifically accurate, I am not sure if they provide us with enough insight into the efficacy of a Neural Operator approach to quantum dynamics. I therefore would not recommend publication unless the authors can make a strong argument for the scalability of their approach or if they provide a more detailed analysis of the limitations of their approach.

It would be appreciated if the authors provided code to reproduce their results.

**Author Comment R3.0:**
We thank the Referee for their thorough review and for drawing attention to issues of scalability and practical applicability. These points have been adequately addressed in the revised manuscript. We now include large-scale experiments for the Hamiltonian Observable approach and a discussion on the limitations of using the entire $2^n$ wavefunction as input. The code to reproduce the results is available at `https://github.com/fs1132429/FNO_for_Quantum`.

(A) Major points:

1.) Where exactly does this method fit in the landscape of numerical approaches? The case where we have access to all $2^n$ amplitudes seems to have limited applicability. It seems that extrapolating beyond the capabilities of other approaches may be a powerful approach, but I am not sure if this work provides evidence that this can be done given the small scale of the experiments.

**Author Comment R3.A1:**
We agree with the Referee's comment that $2^n$ amplitudes as inputs has limited applicability. Thus we have added an adequate discussion to address this in Section 2. We have also performed large-scale experiments on the hamiltonian observables i.e., the expectation value of the observables as inputs. We now present results for both 8-qubit and 20-qubit systems (Figure 4). The larger 20-qubit system demonstrates that the FNO can scale to substantially larger quantum many-body systems, addressing the original concern about the method being limited to small sizes. The new discussion emphasizes that the FNO maintains accurate extrapolation performance as system size increases. Specifically, we report relative errors of 6.46% for 8 qubits and 5.8% for 20 qubits for predictions beyond the training time interval. Furthermore, the increase in training data is modest (from 18,000 to 22,500 samples), demonstrating the viability of our approach despite the exponential growth of the Hilbert space. We have also added explicit details in Section 1C describing how the time-evolved expectation-value data is generated for the 20-qubit system, providing strong evidence of the method's scalability and performance across larger system sizes.

New discussion on limitations in Section 2:
*However, this approach does not scale efficiently, since the Hilbert space dimension of the wavefunction increases exponentially with the number of qubits. Nevertheless, it remains useful for faster evaluation as, for an 8-qubit system, it yields a 6.71$x$ speedup in evaluation compared to exact solvers. Furthermore, because the FNO learns the underlying operator and considers the full wavefunction as input, it can reliably extrapolate to approximately 10$x$ the training time, a regime that is computationally demanding for exact solvers and inaccessible to tensor-network methods due due to increasing entanglement*

New discussion in Section 1C:
*For our experiments, we consider system sizes of 8 and 20 qubits. For the larger system (20 qubits), the time-evolved expectation-value data is generated using the `CUDA-Q` library with Trotterization, since exact simulation at these scales is computationally intractable.*

Old Figure 4:

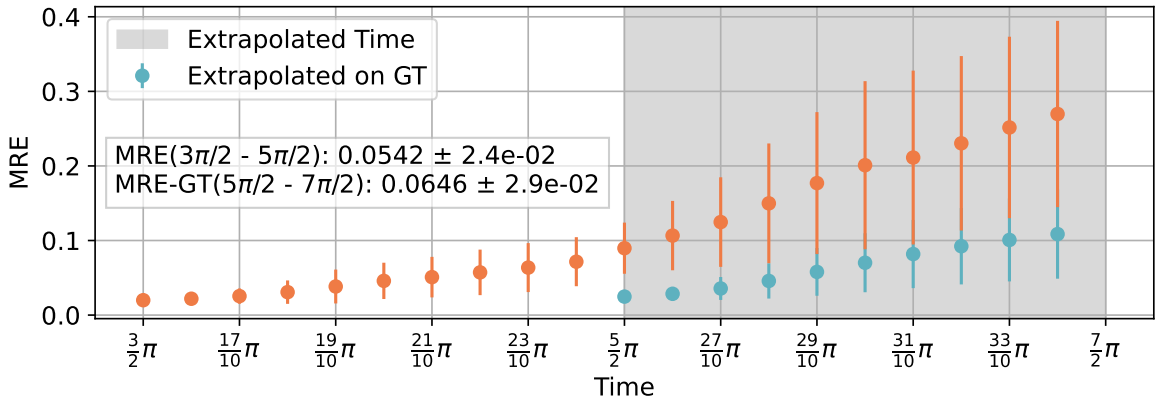
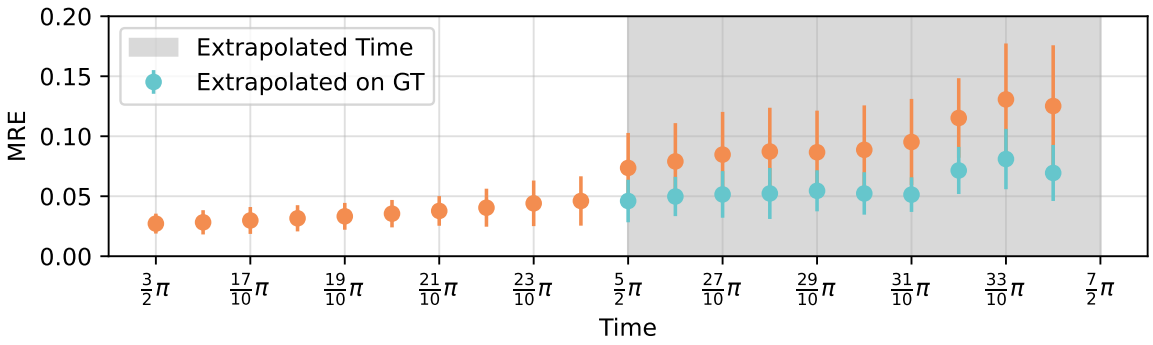



Figure 22: Mean Relative Error (MRE) for predicted Hamiltonian observables in an 8-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. For extrapolated time predictions on ground truth (GT), the MRE is 6.46%. This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 11.18% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $3.096 \times 10^{-6} \pm 2.497 \times 10^{-6}$ and for future time predictions over the same interval on ground truth as $4.212 \times 10^{-6} \pm 3.333 \times 10^{-6}$.

New Figure 4:

**(a)**




**(b)**

| **n** | MRE($24\pi/10$) | MRE − GT($5\pi/2$) | MRE − GT($31\pi/10$) | MRE − GT($34\pi/10$) |
|---|---|---|---|---|
| 8 | $0.0675 \pm 0.0305$ | $0.0242 \pm 0.0114$ | $0.0815 \pm 0.0484$ | $0.1057 \pm 0.0573$ |
| 20 | $0.0460 \pm 0.0208$ | $0.0459 \pm 0.0176$ | $0.0515 \pm 0.0150$ | $0.0704 \pm 0.0247$ |

Figure 23: **(a)** Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. The MRE is 5.8% over the extrapolated time predictions on ground truth (GT). This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 9.6% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $2.16 \times 10^{-9} \pm 2.15 \times 10^{-9}$ and for future time predictions over the same interval on ground truth as $5.34 \times 10^{-9} \pm 4.13 \times 10^{-9}$. **(b)** The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ for systems with 8 and 20 qubits for hamiltonian observables.

Original discussion in Section 1C:
*As seen in Figure 4, with a system of 8 qubits we predict future observables with a relative error of* 6.46%, *even when extending the prediction to double the train time horizon on ground truth.*

New discussion in Section 1C:
*As seen in Figure 4, the FNO maintains consistent and accurate performance even as the system size increases substantially. For 8- and 20-qubit systems, we predict future observables with relative errors of* 6.46% *and* 5.8%, *respectively, even when extending the prediction to twice the training time horizon on ground truth data. The model exhibits no degradation in accuracy with increasing system size, and its extrapolation performance remains stable across all settings. This strongly demonstrates that FNOs can scale effectively to larger quantum many-body systems while preserving error rates in both the training regime and long-time extrapolation. It is also noteworthy that the required volume of training data does not scale in proportion to the exponential growth of the Hilbert space. For instance, we use 18,000 training samples for the 8-qubit system and only 22,500 for the 20-qubit system. This modest increase, despite the dramatic growth in system dimensionality, underscores the data efficiency of the FNO architecture and further illustrates its practical scalability for large-scale quantum dynamical prediction.*

2.) "To enhance the learning process, we incorporate a position embedding Embedstate(k) = k/2n into the input channel dimension, where k is the index associated with each basis state. This approach helps

the model better understand the relationship between the states and their corresponding energy levels." I am very curious about this architectural choice. Is this a common technique in this area? Does the model performance suffer significantly if one does not do this?

**Author Comment R3.A2:**
We thank the Referee for the comment. The position embedding $\text{Embed}_{\text{state}}(k) = k/2^n$ assigns a normalized index to each computational basis state, providing the FNO with explicit information about the ordering of states. An adequate discussion has been added in the main text.

Original discussion in Section 1B2:
*This approach helps the model better understand the relationship between the states and their corresponding energy levels.*

New discussion in Section 1B2:
*In this setting, the ordering of the basis states corresponds to increasing energy levels. By providing this positional embedding, the FNO can more effectively capture the hierarchical structure of the wavefunction, distinguishing between low- and high-energy components. This is particularly relevant because the FNO operates in Fourier space, where high-frequency modes are truncated. The position embedding ensures that these truncated modes predominantly correspond to high-energy components that contribute minimally to the system's dynamics, thereby enhancing both the learning efficiency and extrapolation accuracy.*

3.) How are the low energy states created? How many modes are there?

**Author Comment R3.A3:**
In the revised manuscript, we now clarify how the low-energy states are generated. A low-energy state is a random superposition supported only on the lowest-energy $M$ computational-basis modes. In the experiments reported in this work, we use $M = 4$ modes for the 4-qubit system and $M = 50$ modes for the 8-qubit system. This clarification, along with the explanation of the procedure, has now been added to the Section III of the manuscript.

New Low-energy states Discussion in Section III:
*We generate the low-energy states referenced in Section IB by computing the energies of all $2^n$ computational-basis states under the system Hamiltonian and select the $M$ configurations with the lowest energies. A low-energy wavefunction is constructed by assigning complex amplitudes only to these $M$ modes, sampled independently from a uniform distribution, while all remaining basis-state amplitudes are set to zero. The resulting state is normalized to have unit $L^2$ norm. In our numerical settings, we use $M = 4$ for the 4-qubit system and $M = 50$ for the 8-qubit system.*

4.) Why is the performance on random wave functions so good? Should it not be harder to learn unstructurered data from a complexity point of view?

**Author Comment R3.A4:**
We thank the Referee for the comment. We have added a discussion in Section 1B1 explaining why the FNO performs better on random wavefunctions compared to low-energy states. This clarifies why training on random inputs enables excellent extrapolation, despite these states being unstructured from a complexity perspective.

New discussion in Section 1C:

*FNO performs better on random wavefunctions as these states already span a wide range of energy modes and configurations. Each random state effectively provides the model with information across the entire Hilbert space, allowing it to learn the dynamics for all relevant energy regions in a single step. In contrast, low-energy states are initially localized to only a small subset of modes, and accurately predicting their future evolution requires the model to capture how population spreads to higher-energy modes over time. This highlights why VTI is necessary for accurately extrapolating low-energy states, whereas random states achieve satisfactory results even when the model is trained on a single time step.*

5.) In the same vein. "Importantly, in this study, we focus on using random wave function states to calculate the expectation values of these observables, making the learning and prediction tasks for the FNO maximally general and, as a result, challenging." Does this not contradict the results of figure 3, which indicates that the FNO performs better on random states?

**Author Comment R3.A5:**
We thank the referee for pointing this typo out. The generality and challenge pertain to the random Hamiltonian coefficients ($J_z$, $h$), not the wavefunction states. The manuscript has been updated to clarify this point.

Original discussion in Section 1C:
*To evaluate the FNO's capabilities, we first considered the Ising Hamiltonian as described in Eq. (2), where $J_z$ and $h$ are randomly generated. Importantly, in this study, we focus on using random wavefunction states to calculate the expectation values of these observables, making the learning and prediction tasks for the FNO maximally general and, as a result, challenging.*

New discussion in Section 1C:
*To evaluate the FNO's capabilities, we first considered the Ising Hamiltonian as described in Eq. (2), where $J_z$ and $h$ are randomly generated. This introduces variability in the Hamiltonian parameters, ensuring that the learning task is general with respect to the Hamiltonian and hence challenging. Importantly, in this study, we focus on using random wavefunction states to calculate the expectation values of these observables*

6.) I do not understand the ground truth extrapolation. Does this mean that you extrapolate from GT at 5/2pi to 26/10pi and calculate the MRE there? And then from 26/10pi to 27/pi again?

**Author Comment R3.A6:**
For the observable-based input, we use the time-domain architecture, where the input consists of observables measured over a given time interval. The FNO then extrapolates the dynamics from this input interval to the subsequent interval. Specifically, the input time interval is $[\frac{3}{2}\pi, \frac{5}{2}\pi]$, and the model predicts the observables over the extrapolated interval $[\frac{5}{2}\pi, \frac{7}{2}\pi]$. The mean relative error (MRE) is computed at each time step within the extrapolated interval.

7.) Does the speedup you mention in the conclusion take the training into account? How does the training scale as a function of system size?

**Author Comment R3.A7:**
We thank the referee for bringing this up. We have now added the training time for the wavefunction-input speedup in Section 1B2. We have also included an analysis of the speedup for Hamiltonian-observable inputs in 20-qubit systems, demonstrating a remarkable order-of-magnitude improvement of approxi-

mately $10^4\times$ in Section 1 C. Additionally, we discuss the training time per epoch and how it scales when comparing 8- and 20-qubit systems.

New discussion in Introduction:
*For Hamiltonian-observable inputs, the FNO demonstrates an even more remarkable inference speedup on the order of $10^4x$ for 20-qubit systems, highlighting its efficiency for larger-scale quantum simulations.*

New discussion in Section 1B2:
*This speedup does not take into account the training cost, which for a moderate size of 8 qubits is approximately 2s per epoch over 200 training epochs.*

New discussion in Section 1C:
*Once trained, FNO enables extremely fast inference of quantum dynamics. For a representative example involving 1500 time steps, FNO inference requires 7.1 s in total, whereas GPU-accelerated Trotterized time evolution implemented in CUDA-Q requires approximately 85 s per time step on a Grace Hopper system and 180 s per time step on an NVIDIA A100 GPU. This corresponds to an inference-time speedup of approximately $1.8\times10^4$ (Grace Hopper) to $3.8\times10^4$ (A100) compared to numerical time-evolution methods. Although this does not include the training cost of the FNO, this cost is incurred only once. For the largest system considered (20 qubits), training requires approximately 40 s per epoch over 1700 epochs, corresponding to a total training time of about 19 hours using an A100 GPU. This cost is substantially reduced for smaller systems, such as 8 qubits, where the training time per epoch is approximately 22 s. While training the FNO entails a substantial one-time computational cost that grows with system size, this cost can be amortized across multiple subsequent inference tasks, such as computing dynamics for new initial states. In contrast, numerical time-evolution methods must be rerun independently for each such trajectory, resulting in a computational cost that grows linearly with the number of simulations.*

8.) Do the authors believe that the an FNO approach could be accurate enough to study new physics? Could an FNO approach, applied at scale provide accurate enough computation of observables to study novel physical phenomenon?

**Author Comment R3.A8:**
We thank the reviewer for this important question. An adequate discussion addressing the potential of the FNO to accurately compute observables and study novel physical phenomena has been added in Section 2.

Original discussion in Section 2:
*Under this architecture, the FNO could feasibly be trained using measurements from noisy quantum devices or data from shallow tensor networks, then infer to later time-scales that are otherwise out of reach of contemporary techniques as they would require either more coherent quantum computers or intractably large tensor networks.*

New discussion in Section 2:
*Notably, our results demonstrate that the FNO maintains consistent performance as system size increases substantially. As shown in Figure 4, for 8- and 20-qubit systems, the FNO predicts future observables with relative errors of 6.46% and 5.8%, respectively, even when extrapolating to twice the training time horizon on ground truth data. Furthermore, it requires only a modest increase in the number of training samples to scale efficiently to larger quantum many-body systems.*
*These results establish FNO as a promising computational framework for exploring novel quantum phenomena and simulating complex quantum many-body systems. Its favorable scalability, capacity for long-*

*time extrapolation, and data-efficient learning enable access to regimes that are challenging for conventional numerical approaches. In particular, the FNO could be explored as a data-driven framework to learn the dynamics of a broad class of systems, including interacting spin chains, fermionic lattice models, and disordered quantum systems that are otherwise computationally intractable. Within this architecture, the FNO may be trained on measurement data obtained from noisy intermediate-scale quantum devices or on data generated by shallow tensor-network simulations, and subsequently leveraged to predict system dynamics at later time scales. Such predictions would otherwise necessitate either highly coherent quantum hardware or tensor-network representations of prohibitive bond dimension*

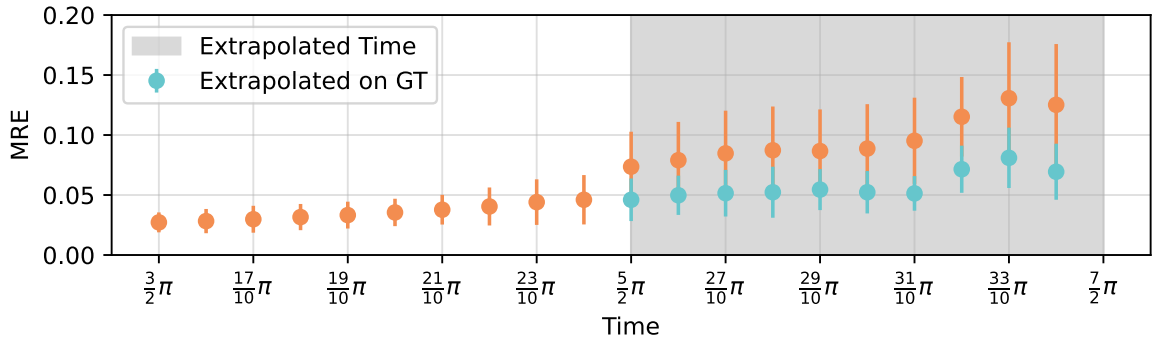
(B) Minor points:

1.) Figure 4 caption, [2/2pi,7/2pi] should be [5/2pi,7/2pi]?

**Author Comment R3.B1:**
We thank the referee for pointing out this typo. The caption of Figure 4 has been corrected, and the extrapolated interval is now given as $[\frac{5}{2}\pi, \frac{7}{2}\pi]$.

New Figure 4:

**(a)**




**(b)**

| **n** | MRE($24\pi/10$) | MRE − GT($5\pi/2$) | MRE − GT($31\pi/10$) | MRE − GT($34\pi/10$) |
|---|---|---|---|---|
| 8 | $0.0675 \pm 0.0305$ | $0.0242 \pm 0.0114$ | $0.0815 \pm 0.0484$ | $0.1057 \pm 0.0573$ |
| 20 | $0.0460 \pm 0.0208$ | $0.0459 \pm 0.0176$ | $0.0515 \pm 0.0150$ | $0.0704 \pm 0.0247$ |

Figure 24: **(a)** Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval $[0, \frac{3}{2}\pi]$ as shown in Figure 1 d) and predicts the output interval $[\frac{3}{2}\pi, \frac{5}{2}\pi]$. Additionally, it extrapolates future unseen dynamics $[\frac{5}{2}\pi, \frac{7}{2}\pi]$, which is twice the length of the output interval. The MRE is 5.8% over the extrapolated time predictions on ground truth (GT). This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is 9.6% denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as $2.16 \times 10^{-9} \pm 2.15 \times 10^{-9}$ and for future time predictions over the same interval on ground truth as $5.34 \times 10^{-9} \pm 4.13 \times 10^{-9}$. **(b)** The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ for systems with 8 and 20 qubits for hamiltonian observables.

---

Report of the First Referee – COMMSPHYS-24-1031

---

(0) I am pleased to see that the authors have taken the feedback from all referees, adding necessary revisions to the manuscript; in particular, the addition of the 20 qubit results address the major concerns with scalability.

Aside from the minor comments below, I feel that the revised manuscript has met the necessary acceptance criterion.

**Author Comment R1.0:**
We thank the referee for their support of the revised manuscript and for acknowledging that the addition of the 20-qubit results addresses the scalability concerns raised in the previous round of review. We are pleased that the revisions have met the acceptance criteria. Below, we respond to the remaining minor comments.

(A) Minor Comments:

1. When discussing the temporal embedding details, authors say "positional embedding" twice. Should this not be "temporal embedding?"

**Author Comment R1.B1:**
We thank the referee for pointing this out. The embedding encodes the time index of each state in the input sequence and is therefore more appropriately referred to as a temporal embedding rather than a positional embedding. We have updated the terminology throughout the manuscript to reflect this.

Original discussion:
*Instead of a single time $t = 0$, we then use a time interval as input, i.e.,*

$$\boldsymbol{I} = \left[\mathrm{Embed}_{\mathrm{time}}, S_{[0,\frac{3}{2}T]}\right] \in \mathbb{C}^{(2^n+2)\times m}, \tag{1}$$

*where $m = \frac{3}{2}T/\Delta t$ defines the number of of equidistant time-steps in the interval $[0, \frac{3}{2}T]$. For training, we choose $T = \pi$ and $\Delta t = \pi/10$. For the positional embedding $\mathrm{Embed}_{\mathrm{time}}$, we use a sinusoidal encoding to facilitate the FNO in learning temporal patterns. Given a discretized time grid with $m$ equidistant points, indexed by $l \in \{0, \ldots, m-1\}$, the positional embedding is constructed using a single cosine–sine pair. With two embedding channels and a fixed frequency $\omega = 1$, each time index $l$ is mapped as*

$$\mathrm{Embed}_{\mathrm{time}}(l) = \left[\cos(\omega l),\ \sin(\omega l)\right]. \tag{2}$$

*This results in a positional embedding tensor of shape $(1, 2, m)$, where the two channels correspond to the cosine and sine components evaluated at each time step. By mapping time onto a smooth periodic representation, the embedding allows the model to capture the relative ordering of temporal points while preserving continuity across the discretized interval cite[li2020fourier,FNOPracticalReview].*

New discussion:
*Instead of a single time $t = 0$, we then use a time interval as input, i.e.,*

$$\boldsymbol{I} = \left[\mathrm{Embed}_{\mathrm{time}}, S_{[0,\frac{3}{2}T]}\right] \in \mathbb{C}^{(2^n+2)\times m}, \tag{3}$$

*where $m = \frac{3}{2}T/\Delta t$ defines the number of of equidistant time-steps in the interval $[0, \frac{3}{2}T]$. For training, we choose $T = \pi$ and $\Delta t = \pi/10$. For the temporal embedding $\mathrm{Embed}_{\mathrm{time}}$, we use a sinusoidal encoding to facilitate the FNO in learning temporal patterns. Given a discretized time grid with $m$ equidistant points, indexed by $l \in \{0, \ldots, m-1\}$, the temporal embedding is constructed using a single cosine–sine pair. With two embedding channels and a fixed frequency $\omega = 1$, each time index $l$ is mapped as*

$$\mathrm{Embed}_{\mathrm{time}}(l) = \left[\cos(\omega l),\ \sin(\omega l)\right]. \tag{4}$$

*This results in a temporal embedding tensor of shape $(1, 2, m)$, where the two channels correspond to the cosine and sine components evaluated at each time step. By mapping time onto a smooth periodic representation, the embedding allows the model to capture the relative ordering of temporal points while preserving continuity across the discretized interval cite[li2020fourier,FNOPracticalReview].*

---

Report of the Third Referee – COMMSPHYS-24-1031

---

(0) I want to say that I appreciate the significant improvements the authors made to the manuscript. The exposition is much clearer, and I believe that the results can now be better understood. I also greatly

appreciate the open source code to reproduce the experiments.

**Author Comment R3.0:**
We thank the referee for their positive assessment of the revised manuscript and for acknowledging the improvements in clarity as well as the release of the open-source code accompanying this work.

(1) However, while I appreciate the performance increase over the exact simulation obtained for the larger systems, I am still not completely convinced about the use case of this approach.

It seems that one in most cases of practical interest, one would want to use a more accurate and controlled technique to calculate either the state (e.g. Tensor Networks, Neural quantum states) or observables (e.g. Pauli propagation) for these system sizes, even if it is more expensive. It's interesting by itself that the dynamics studied here can be compressed reasonably well.

**Author Comment R3.1:**
We thank the referee for this comment. We agree that established techniques such as tensor networks, neural quantum states, and Pauli propagation provide accurate and controlled approaches for simulating quantum dynamics in many regimes. Our work instead focuses on learning an approximation to the underlying time-evolution operator from data, which can then be used to efficiently predict dynamics across many initial conditions and extrapolate to later times once trained.

To clarify this point, we have added discussion in the Introduction and Section 2 highlighting regimes where this operator-learning framework may be advantageous, such as when entanglement growth limits tensor-network simulations or when operator spreading increases the cost of Pauli propagation methods.

New discussion in Introduction:
*Rather than directly simulating the evolution for each instance, the FNO learns an approximation to the underlying time-evolution operator from data. This approach can be advantageous in regimes where entanglement growth or operator spreading makes conventional simulation methods increasingly costly cite[shao2026character, Orus2019].*

New discussion in Section 3:
*In contrast to conventional simulation methods, tensor networks become challenging at long evolution times due to entanglement growth, while Pauli propagation can become expensive when operator support spreads broadly across the system cite[shao2026character, Orus2019]. By learning an approximation to the underlying time-evolution operator, the FNO enables efficient prediction across many initial conditions and can extrapolate the dynamics to longer timescales without repeatedly solving the full time-evolution problem.*

(2) While the authors now propose using measurement data to learn the FNO, this seems like a highly non-trivial extension, since one would now require projective measurements for learning instead of the full states, which arguably contain much less information.

**Author Comment R3.2:**
We thank the referee for raising this point. We agree that learning dynamics from measurement data is a non-trivial problem since such data contains less information than full quantum states. In fact, this scenario is closely related to the setting studied in Sec. 1C and Fig. 4 of the manuscript, where the

FNO is trained using expectation values of Hamiltonian observables rather than the full wavefunction. While in this work these observables are computed from simulated quantum states, they correspond to physically measurable quantities and therefore represent a setting where only partial information about the system is available. As shown in Fig. 4, the model is still able to accurately predict future dynamics and extrapolate to longer times when trained on these observable expectation values. We have clarified this point in the manuscript.

New discussion in Section 2C:
*In particular, learning dynamics from such observable data is a non-trivial extension compared to learning from full state representations, since the model must implicitly infer and propagate latent information about the system that is not directly observed.*

New discussion in Section 2C:
*We emphasize that throughout this section the FNO is never provided access to the full quantum state, and instead operates entirely on observable expectation values. This demonstrates that the model can learn and extrapolate quantum dynamics in a setting that is directly compatible with experimentally accessible measurement data.*

(3) I believe the results in this work are correct, but probably not the most impactful for the field of simulating quantum dynamics. I would therefore recommend publication if the impact is not a major concern for the editors.

**Author Comment R3.3:**
We thank the referee for the positive assessment that the results are correct. While established numerical techniques remain the most accurate tools for computing individual trajectories of quantum dynamics, the goal of this work is to investigate whether neural operators can learn an approximation to the underlying time-evolution operator and thereby enable efficient reuse across many initial conditions and parameter regimes. Since the initial release of this manuscript, this approach has already begun to be adopted in subsequent work. In particular, recent work has used Fourier Neural Operators within optimization loops for quantum control design, building directly on the framework introduced here. We have added an adequate discussion in the manuscript. We have also clarified in the revised manuscript that the primary advantage of this method lies not in replacing high-accuracy solvers for single trajectories, but in enabling efficient evaluation in regimes where conventional methods become computationally demanding, such as in the presence of entanglement growth or operator spreading, or when repeated simulations are required.

New discussion in Introduction:
*Rather than directly simulating the evolution for each instance, the FNO learns an approximation to the underlying time-evolution operator from data. This approach can be advantageous in regimes where entanglement growth or operator spreading makes conventional simulation methods increasingly costly cite[shao2026character, Orus2019].*

New discussion in Section 3:
*In contrast to conventional simulation methods, tensor networks become challenging at long evolution times due to entanglement growth, while Pauli propagation can become expensive when operator support spreads broadly across the system cite[shao2026character, Orus2019]. By learning an approximation to the underlying time-evolution operator, the FNO enables efficient prediction across many initial conditions*

*and can extrapolate the dynamics to longer timescales without repeatedly solving the full time-evolution problem.*

New discussion in Section 3:
*Beyond forward simulation, learned surrogates for quantum dynamics may also enable tasks such as parameter exploration and inverse design, where repeated evaluations of time evolution are required. In classical settings, Fourier Neural Operators have been successfully used for inverse design and control in systems governed by partial differential equations, where learned surrogates are embedded within optimization loops to efficiently explore parameter spaces cite[li2023fourier, kovachki2023neural]. Analogously, this approach can be leveraged in quantum settings to enable efficient optimization over control protocols or Hamiltonian parameters. This approach has been adopted in subsequent work building on the framework introduced here, where Fourier Neural Operators are incorporated into optimization loops to design control protocols for quantum systems cite[pipi2025fnoinversedesign]. This development highlights the broader applicability of our approach.*

(A) Minor Comments:

1. It is still not explained what is meant by "Extrapolated on GT" in figure 4. Does that mean you start with the exact state at 5/2pi?

**Author Comment R3.A1:**
We thank the referee for pointing out that this was unclear. In the "extrapolated on GT" setting, the extrapolation interval $[5\pi/2,\ 7\pi/2]$ is initialized using the ground-truth value at $t = 5\pi/2$. The FNO then predicts the subsequent dynamics in this interval. This setup allows us to evaluate the model's ability to extrapolate future dynamics independently of errors accumulated in the preceding prediction interval. We have clarified this point in the caption of Fig. 4.

Original figure 4 caption:
***(a)*** *Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval* $[0, \frac{3}{2}\pi]$ *as shown in Figure 1d) and predicts the output interval* $[\frac{3}{2}\pi, \frac{5}{2}\pi]$*. Additionally, it extrapolates future unseen dynamics* $[\frac{5}{2}\pi, \frac{7}{2}\pi]$*, which is twice the length of the output interval. The MRE is* 5.8% *over the extrapolated time predictions on ground truth (GT). This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the model was trained on, the MRE for extrapolated time predictions is* 9.6% *denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as* $2.16\times10^{-9}\pm2.15\times10^{-9}$ *and for future time predictions over the same interval on ground truth as* $5.34\times10^{-9}\pm4.13\times10^{-9}$*.* ***(b)*** *The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time* $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ *for systems with 8 and 20 qubits for hamiltonian observables.*

New figure 4 caption:
***(a)*** *Mean Relative Error (MRE) for predicted Hamiltonian observables in a 20-qubit system using FNO, utilizing the time-domain architecture. The model is trained on the input time interval* $[0, \frac{3}{2}\pi]$ *as shown in Figure 1d) and predicts the output interval* $[\frac{3}{2}\pi, \frac{5}{2}\pi]$*. Additionally, it extrapolates future unseen dynamics* $[\frac{5}{2}\pi, \frac{7}{2}\pi]$*, which is twice the length of the output interval. The MRE is* 5.8% *over the extrapolated time predictions on ground truth (GT). In this setting, the extrapolation is initialized using the ground-truth value at* $t = 5\pi/2$*, so that the model predicts the subsequent dynamics in* $[5\pi/2, 7\pi/2]$ *without error propagation from the previous prediction window. This is particularly significant in quantum simulations, where the primary objective is to extend these simulations. Additionally, based on the time interval the*

*model was trained on, the MRE for extrapolated time predictions is* 9.6% *denoted by orange dashes. We additionally calculate the mean MSE loss for the training time interval as* $2.16 \times 10^{-9} \pm 2.15 \times 10^{-9}$ *and for future time predictions over the same interval on ground truth as* $5.34 \times 10^{-9} \pm 4.13 \times 10^{-9}$. ***(b)*** *The results present the MRE at specific time steps for both the train time (i.e., the time intervals that FNO encountered during training) and the extrapolated time* $[\frac{5}{2}\pi, \frac{7}{2}\pi]$ *for systems with 8 and 20 qubits for hamiltonian observables.*

New discussion in Section 2C:

*Given that these observables present challenges for the FNO to learn effectively, we evaluate extrapolation in two settings. In the first, the model predicts the interval* $[\frac{3}{2}T, \frac{5}{2}T]$ *from the training data and then continues to extrapolate autoregressively. In the second, which we denote as extrapolation on ground truth (GT), the extrapolation interval* $[\frac{5}{2}T, \frac{7}{2}T]$ *is initialized using the ground-truth value at* $t = \frac{5}{2}T$. *The model then predicts the subsequent dynamics in this interval. This procedure isolates the model's ability to extrapolate future dynamics without compounding errors from the preceding prediction window.*