A quantum model should be like a puppy, so that it could be:

- *Expressive, i.e. smart enough to learn commands like “sit” and “come”, and later “stay”, “jump” and “roll over”*
- *Trainable, i.e. to learn new tricks eagerly with few praises and treats*
- *Stable, so that it responds to you not only at home but also in the park while playing with other dogs*

Introduction
What is expressivity and trainability
How are they related to stability
Why temporal data is special
Different objectives -
different models and criteria
Measuring QTSA
expressivity ↔ trainability
vs. stability
When to stop, how to balance?
Summary and reflection

Expressivity ↔ Trainability vs. Stability
in Quantum Model Development

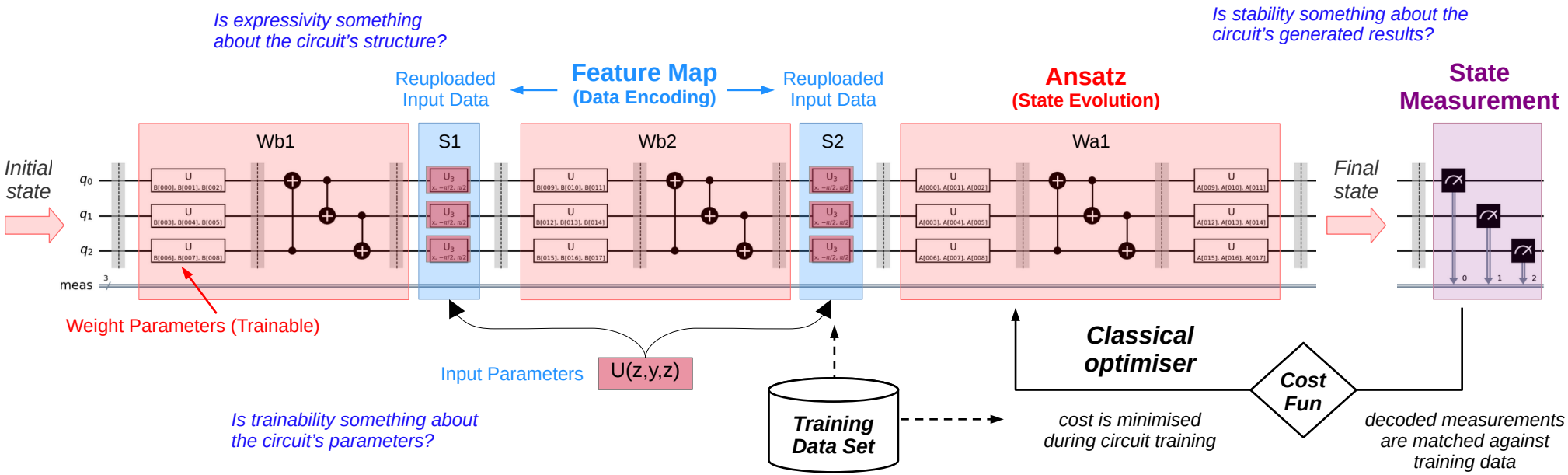
Jacob L. Cybulski
Enquanted, Melbourne, Australia

Sebastian Zając
Military University of Technology, Warsaw, Poland

Episode LXXX of Warsaw Quantum Computing Group meetup, Venue: Warsaw School of Economics, Organized by Fundacja Quantum AI and QPoland, Warsaw, Poland, 27 Aug 2026.

Parameterised Quantum Model (PQC)

Seeking clarity on what we all understand



Input data is encoded into the feature map's input parameters, setting the initial quantum state and optionally altering states by data reuploading later.

Ansatz evolves the quantum state; it consists of layers of trainable blocks with parameterised quantum gates interwoven with entangling blocks.

The circuit final state is measured and decoded (interpreted) as the model's output in the form of classical data (no mid-circuit measurement).

Typically PQCs are trained using hybrid methods of Variational Quantum Algorithms (VQA) involving classical optimisers and classical loss / cost functions

What is *expressivity*
What is *trainability* of a QML model?
What is *stability*

A quantum model should be like a sculptor, who is:
- *expressive, using imagination to create shapes in his mind*
- *trainable, to quickly learn new tools and carving techniques*
- *stable, as he adapts to work with wood imperfections*

Expressivity
is the model's capacity to represent complex data within the ***quantum space***.

What could it mean?
- How many different functions, approximating a given dataset, can the model accommodate?
- Is this model of the right size for this specific dataset?
- What is the upper bound of the model's learning capacity?
- What is the performance for a given dataset and each combination of weight parameters?
- Will gradient descent actually find an optimum solution?
- Can this circuit reach anywhere in the Hilbert space?
- What is the boundary of states this circuit can explore?

↔

Trainability
is the model's navigability—how easily an optimiser can find the best configuration within the model *parameter space*.

What could it mean?
- What is the step-by-step progress of the training process?
- At what point the model overfits data?
- Is the model structure well-designed to ensure its optimisation convergence?
- Are there any sharp ravines or flat valleys that will stall the classical optimiser? (barren plateaus)
- Is the model structurally constrained to avoid barren plateaus?
- Is there a specific training configuration causing vanishing / exploding gradients?
- Can the state-space geometry be inverted for the natural gradient step?
- Will the natural gradient advantage vanish as the system scales?

vs.

Stability
is the model's consistency in producing reliable results, despite minor changes to its data or operating environment.

What could it mean?
- Is the model robust against small shifts or noise in the classical input data?
- How sensitive is the trained model to small parameter shifts or hardware errors?
- Does the model maintain its performance on unseen population data?
- Is the model aligned with the target function across data subsets?
- How rapidly does the model's accuracy degrade in the presence of physical hardware noise?

Blue *indicates concerns that are explicitly regarding quantum phenomena, whereas* ***Magenta*** *highlight issues of classical parameters, and* ***Black*** *is either.*

Why don't we see any ***Magenta*** *colour here?*

Jacob L. Cybulski and Sebastian Zając (2026): "Quantum Modelling of Time Series: Expressivity vs. Trainability."
Proc. of Int. Conf. on Quantum Comp. Infrastructure and Technologies (QUANCOM 2026), 1-4 September, Trento, Italy.

Hybrid quantum-classical VQA process

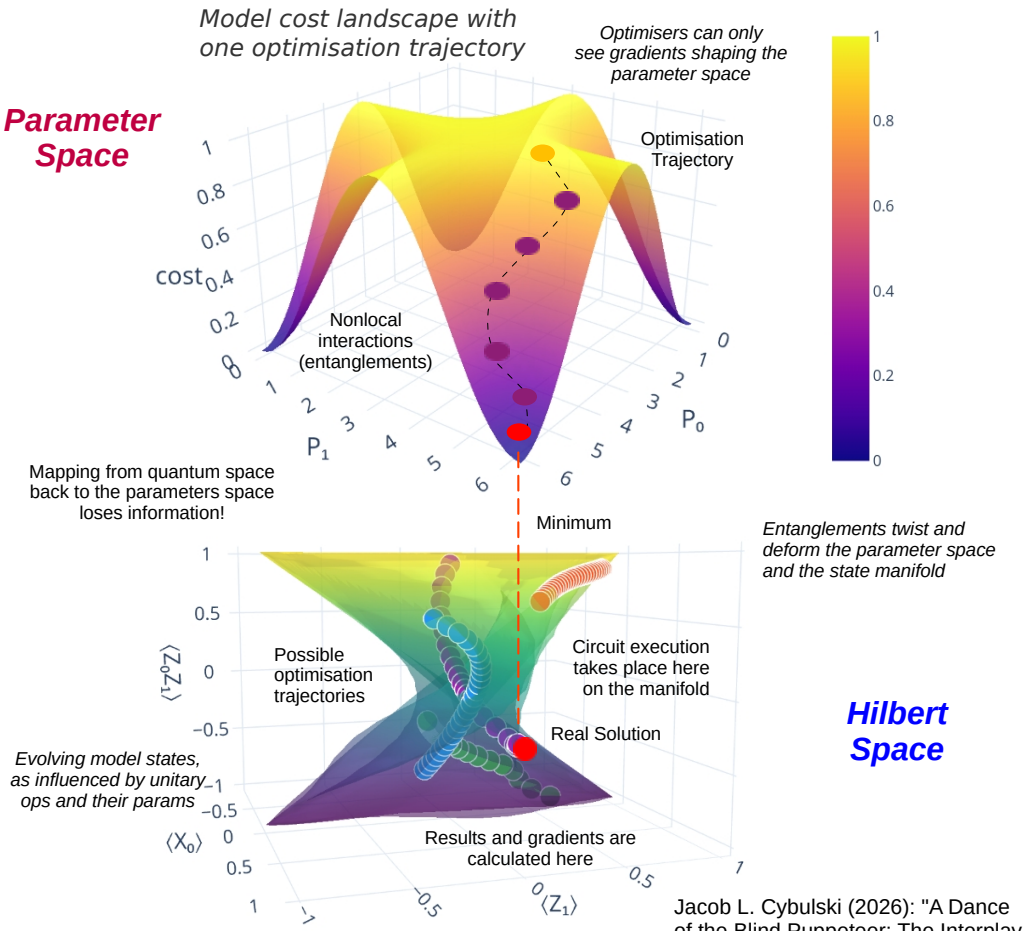
Parameter space
(dim = the number of params) is a classical multi-dim space of trainable gate parameters, which the optimiser navigates.

Classical parameters
A classical optimiser can only manipulate the gate parameters but is not aware of their action on qubits states or their correlations via qubit entanglements.

Measurement
of individual qubits collapses their states, projecting the circuit state onto classical outcomes - the only information reaching the optimiser. In the process we lose some quantum information (e.g. phase).

Sebastian Zając. Jacob L. Cybulski, Bartosz Dziewit and Tomasz Kulpa (2026): "The Inverse Born Rule Equivalence. On the Informational Limits of Real-Valued Amplitude Encodings and the Measurement of Quantum Advantage in Data Embeddings." arXiv arXiv:2602.21350.

Expressivity, trainability and stability problems: what are their *quantum* and *classical* causes?



Warped model manifold with four classical optimisation trajectories within the Hilbert state space

Jacob L. Cybulski (2026): "A Dance of the Blind Puppeteer: The Interplay Between a Classical Optimizer and the Hilbert Space." in Zapata (Ed), A Portrait of Quantum Technologies in Finance, TQFB, 107-121.

Hilbert state space
(dim ≈ $2^{\text{the number of qubits}}$) is the quantum realm where the models and their states evolve in response to unitary operations as defined by the circuit gates.

Data encoding
brings in classical data into the Hilbert space as unique and correlated quantum states during the model execution. Bad encoding prevents effective model optimisation.

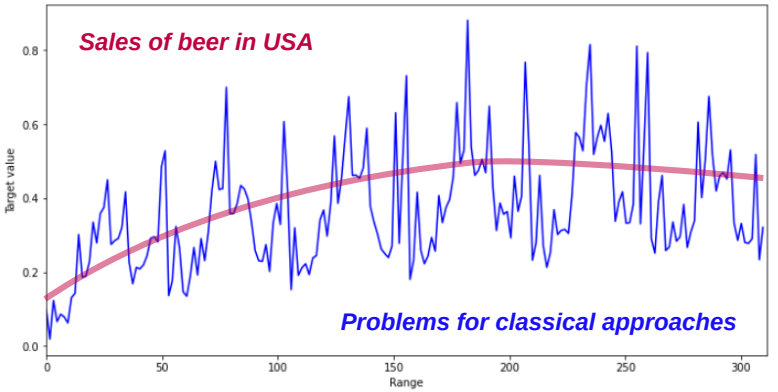
Entanglements
create non-separable qubit states, which cause non-local interactions and warp the state space geometry. As a side-effect, they deform the cost landscape, affecting the optimisation process.

Circuit layers of gates
determine the evolution of the model's initial into the final state during execution (forming a state space trajectory).

QTSA
Why QML in time series (TS)

- Classical approaches to time series analysis are highly effective, i.e., statistical, ML and deep learning!
- They still struggle with long-term and highly complex temporal patterns, volatility, chaotic behaviour, etc.
- Quantum time series analysis targets these problem areas and has been successful in some applications.

- Our team developed a number of distinct QTSA models:
  - ***Quantum curve-fitting models (QCF):*** fit data values at specific points in time to a function.
  - ***Quantum forecasting models (QFC):*** predict future values from a sliding window (SW) of their historical context.
  - ***Quantum denoising autoencoders (QDAE):*** compress noisy input into a latent space, then decompress into noise-reduced output, while noise is redirected into trash space.
  - ***Quantum reservoir computing (QRC):*** predict future values based on the incoming data and own memory.



- What quantum model characteristics can positively affect performance of time series analysis?
  - Randomness of observable results ***(measurement of quantum states)***
  - Considering alternative decisions concurrently ***(superposition of quantum states)***
  - Controlling parallel choices with constraints ***(entanglement of quantum systems)***
- What time series characteristics can negatively affect quantum model performance?
  - Same issues as in classical analysis, and especially…
  - Auto-correlation of features (temporal dependencies)
  - Feature homogenisation (neighbouring values are similar)
  - Time series volatility, noise and anomalies
  - Temporal obsolescence (old values not useful), plus...
  - Quantum model's lack of persistent memory
  - Classical optimiser lack of awareness of quantum phenomena

Cuéllar, M.P., Pegalajar, M.C., Ruiz, L.G.B., Cano, C.: Time Series Forecasting with Quantum Neural Networks. In: Rojas, I., Joya, G., Catala, A. (eds.) Adv. Comput. Intell. pp. 666–677. Springer Nature Switzerland (2023)

A quantum model should be like a racing car, with its:

- *expressivity as the power of its engine and breaks*
- *trainability as the ease of steering around the corners*
- *stability ensuring it does not fly off the road at speed*

Recent Research

Focus on models -
what is the best quantum model architecture?

The concepts of expressivity and trainability,
as well as their principles and causes were adopted from the literature.

Methods and Data

Methods, for Curve-Fitting and Forecasting models...

- 3 curve-fitting (CF), 3 forecasting (FC, incl. one classical) model types.
- Explored models by changing the qubits / layers numbers, and their roles.
- Generated 50 model architectures, such as:
  CF: 5 Serial, 16 Parallel, 5 Xparallel; FC: 14 Xqnn, 6 OvLoad, and 4 Cnn.
- Created 10 differently initialised instances of each model (500 in total).
- Implemented in Qiskit, using EstimatorQNN primitives, trained with NeuralNetworkRegressor, measured expectation values of qubits.
- Optimised with L_BFGS_B + MSE cost function, scored with MSE and R2 metrics on training and test data partitions.
- R2 was used to compare across CF / FC model types.
- Reported median + IQR performance of model instances.
- Noise-free simulation was used to avoid confounding factors.

Data preparation, for...

- Time series “2-sins” was synthetic (simple + repetitions).

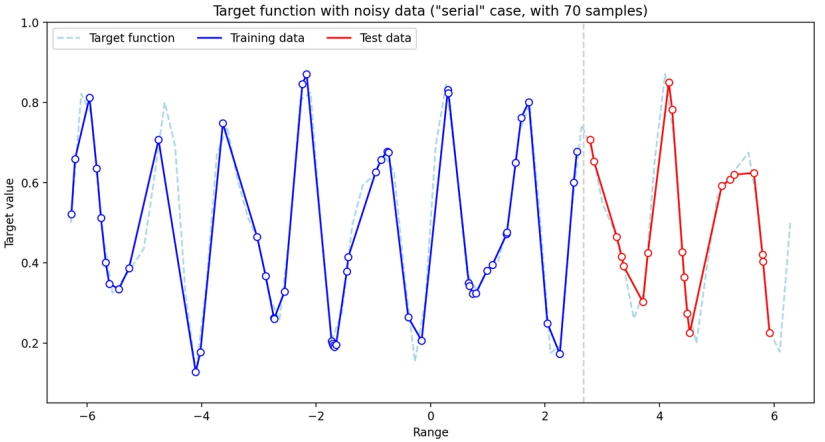
  CF: TS was split into 50 + 20 points for training + testing.

  FC: 70 data points formed 65 sliding windows of 5 data points each, offset by 1 step, horizon of 1, incomplete windows discarded.
- The series was split into 46 + 19 windows for training and testing.
- Training and testing sets had an approximate ratio of 71/29.

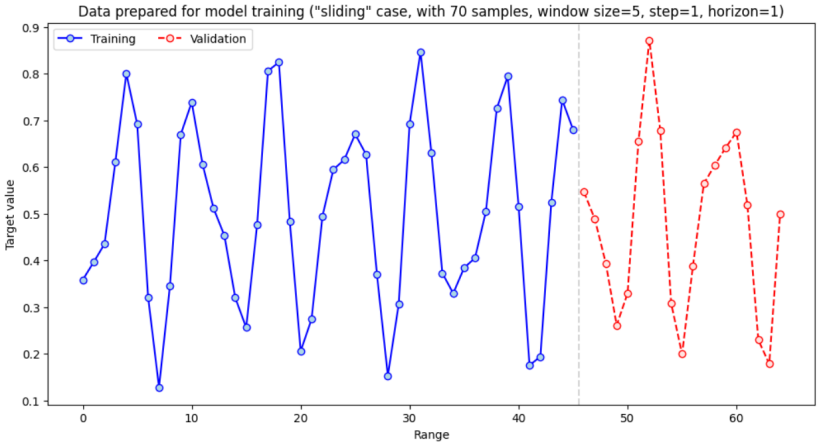
Function *2-sins:*

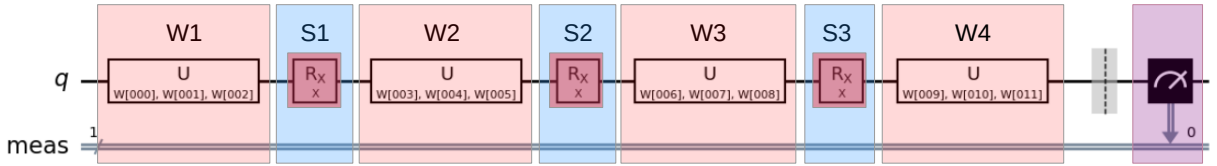
$$f_{2sins}(x) = \frac{sin(5.0 * x) + 0.5 * sin(8.0 * x)}{4} + 0.5$$

Function *2-sins* sampled for curve-fitting models:



Function *2-sins* sampled for sliding-window forecasting models:



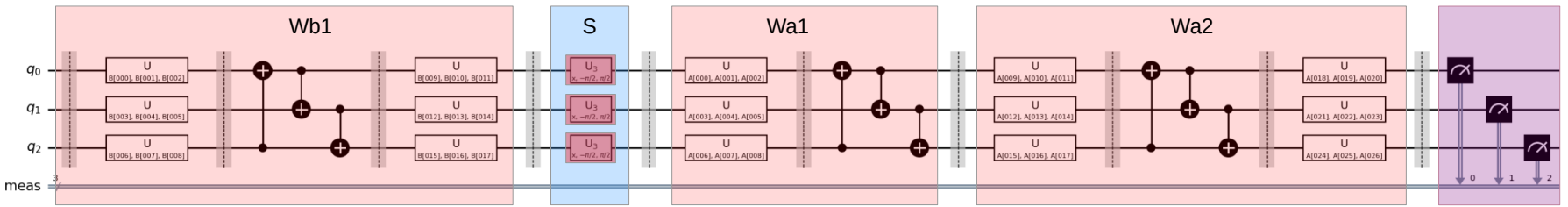


Different structures & functions / Different strengths & weaknesses

CF Serial

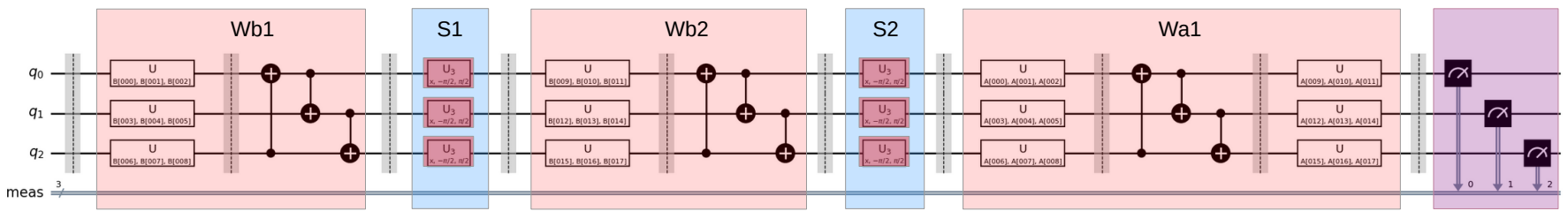
- Serial reuploading + trainable layers → high-dimensional parameter-space → high expressivity in 1 qubit state space
- Hilbert space bounded at 2D → immune to barren plateaus → fast and effective training → but unstable in tests (IQR)
- Deep circuits, lots of parameters → spectral dilution ≡ thin gradient across Fourier frequency spectrum

CF Parallel



- Parallel reuploading + trainable layers → we found they had poor expressivity → little effect of trainable layers
- Training suggested high stability (IQR=0) → instead hard learning capacity ceiling → sub-optimal tests (high IQR)

CF Extended Parallel

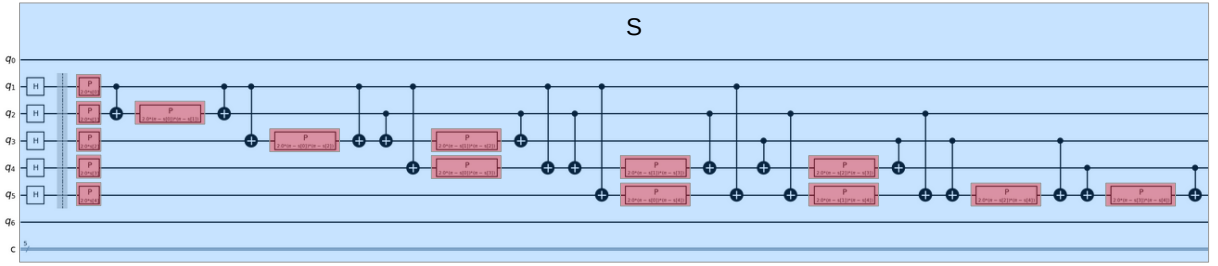


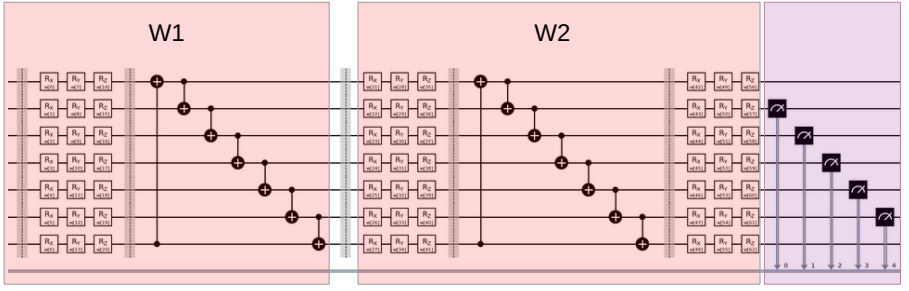
- Parallel + serial reuploading + trainable layers → high expressivity
- Best models: few qubits + deep circuits + not many parameters → superior training and test performance!
- Serial reuploading compensates reduced-dimensionality of the Hilbert space

Curve-fitting models

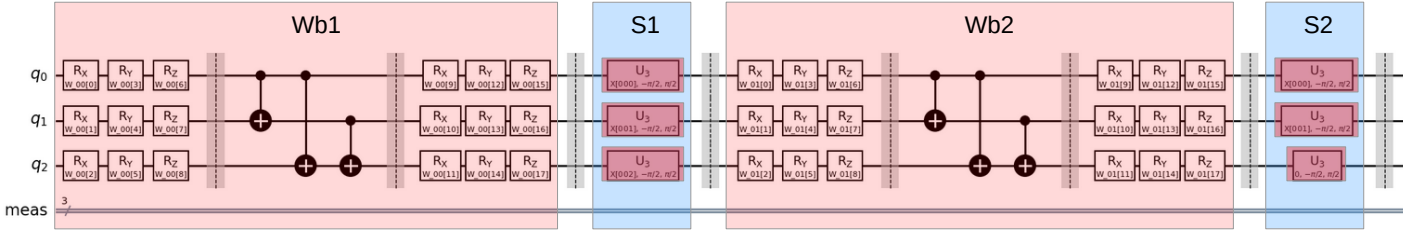
Different structures & functions / Different strengths & weaknesses

- ZZ feature map → highly expressive data encoding
- Auxiliary qubits → higher-dimensional Hilbert space → more parameters + entanglements → highly expressive model → rapid overfitting
- However, high-dimensional Hilbert space + high entanglement → homogenisation of close TS values → scrambling of closely related states → unstable tests (high IQR).

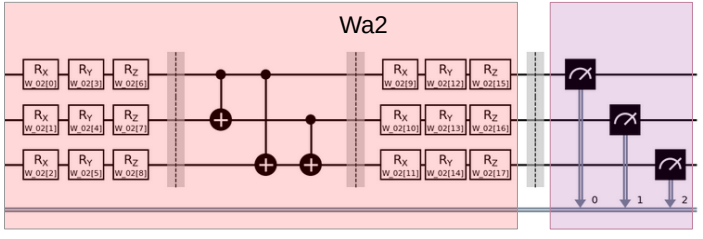




FC Extended QNN
with Havlíček ZZ feature map



FC Overloading QNN
with structural regularisation

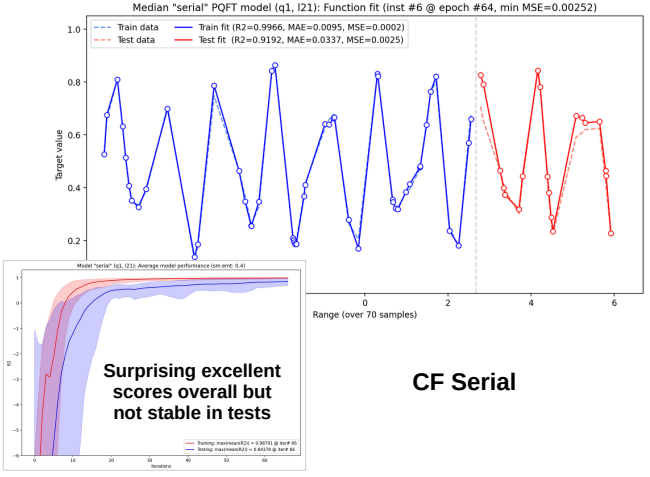


- Overloading of qubit function + serial data reuploading → qubit reuse + highly expressive data encoding
- Prefers high qubits number + high parameters number → structural regularisation → steady performance gain → most stable (low IQR) + highest test performance (R2) *(among quantum forecasting models)*

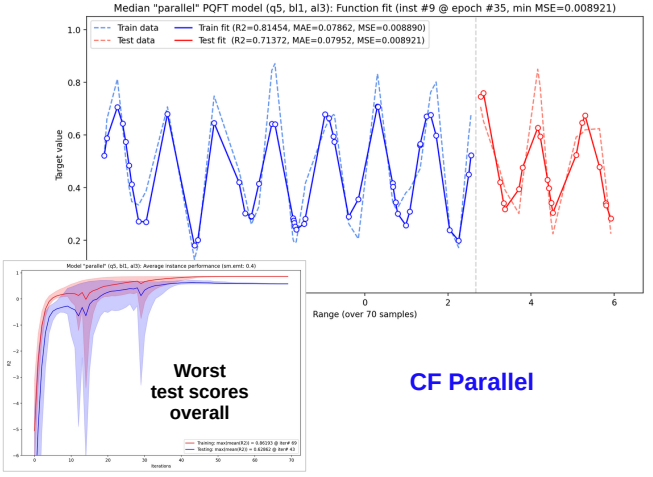
Forecasting models

Training / Testing, Median R^2 Scores, and Data Fit

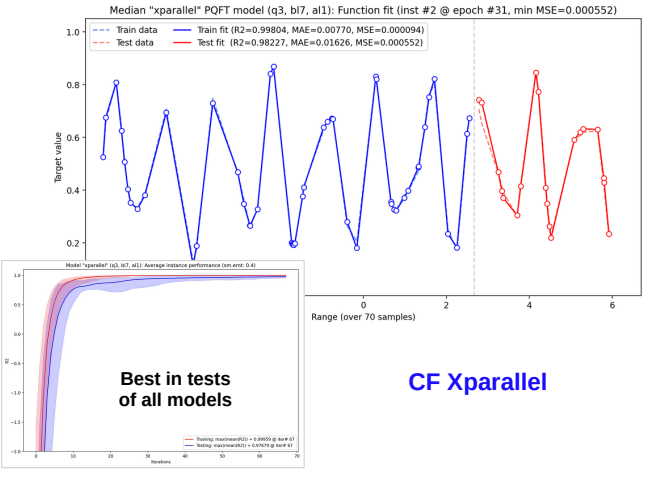
(best configurations selected on test partition as shown)



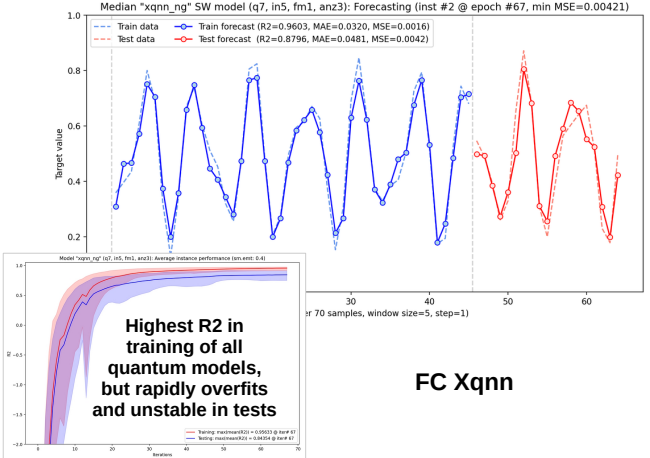
CF Serial



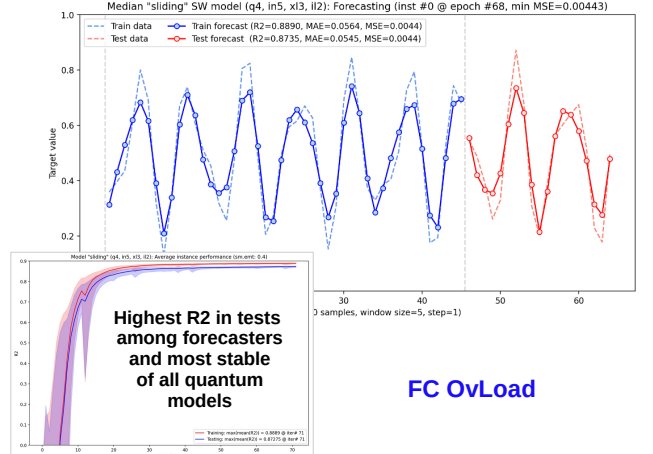
CF Parallel



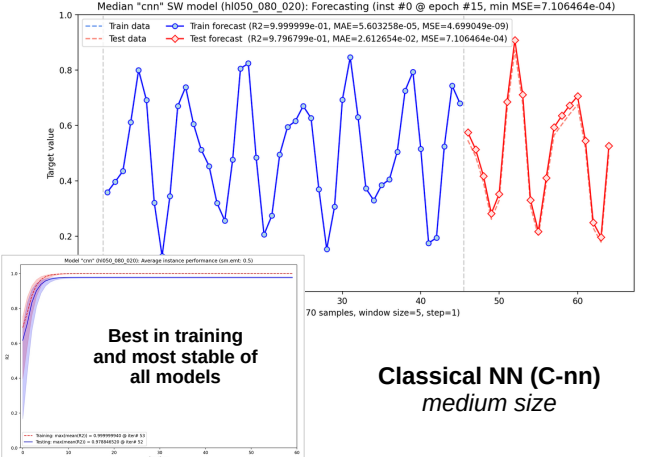
CF Xparallel



FC Xqnn



FC OvLoad



Classical NN (C-nn)
medium size

Parallel vs. Xparallel

curve-fitting models

CF Parallel

Model	Type	Qubits	Params	Training R2 (IQR)	Testing R2 (IQR)	Training MSE (IQR)	Testing MSE (IQR)	Specs
parallel	CF	2	48	0.0588 (0.0000)	-0.1407 (0.0391)	0.0451 (0.0000)	0.0355 (0.0012)	q2 bl3 al3
parallel	CF	3	72	0.1088 (0.0000)	-0.1652 (0.1275)	0.0427 (0.0000)	0.0363 (0.0040)	q3 bl3 al3
parallel	CF	4	96	0.1096 (0.0000)	-0.2544 (0.1597)	0.0427 (0.0000)	0.0391 (0.0050)	q4 bl3 al3
parallel	CF	5	120	**0.8620** (0.0000)	0.6939 (0.0283)	**0.0066** (0.0000)	0.0095 (0.0009)	q5 bl3 al3
parallel	CF	5	60	0.8604 (0.0082)	0.7079 (0.0297)	0.0067 (0.0004)	0.0091 (0.0009)	q5 bl1 al1
parallel	CF	5	120	**0.8620** (0.0000)	0.6939 (0.0283)	**0.0066** (0.0000)	0.0095 (0.0009)	q5 bl3 al3
parallel	CF	5	180	0.8620 (0.0000)	0.6816 (0.0351)	**0.0066** (0.0000)	0.0099 (0.0011)	q5 bl5 al5
parallel	CF	5	240	0.8620 (0.0000)	0.6690 (0.0310)	**0.0066** (0.0000)	0.0103 (0.0010)	q5 bl7 al7
parallel	CF	5	60	0.8604 (0.0082)	0.7079 (0.0297)	0.0067 (0.0004)	0.0091 (0.0009)	q5 bl1 al1
parallel	CF	5	90	0.8620 (0.0001)	**0.7109** (0.0374)	**0.0066** (0.0000)	**0.0090** (0.0012)	q5 bl1 al3
parallel	CF	5	120	0.8620 (0.0000)	0.6959 (0.0240)	**0.0066** (0.0000)	0.0095 (0.0007)	q5 bl1 al5
parallel	CF	5	150	0.8620 (0.0000)	0.6989 (0.0249)	**0.0066** (0.0000)	0.0094 (0.0008)	q5 bl1 al7
parallel	CF	5	90	0.8620 (0.0001)	**0.7109** (0.0374)	**0.0066** (0.0000)	**0.0090** (0.0012)	q5 bl1 al3
parallel	CF	5	120	**0.8620** (0.0000)	0.6939 (0.0283)	**0.0066** (0.0000)	0.0095 (0.0009)	q5 bl3 al3
parallel	CF	5	150	0.8620 (0.0000)	0.7013 (0.0246)	**0.0066** (0.0000)	0.0093 (0.0008)	q5 bl5 al3
parallel	CF	5	180	0.8620 (0.0000)	0.6877 (0.0299)	**0.0066** (0.0000)	0.0097 (0.0009)	q5 bl7 al3

The illusion of stability

Is this model (left) efficient and stable?

"stable" training R2/MSE +
very low training IQR +
poor testing scores and IQR

Repeated layers of added parameters and entanglements, either pre- or post-encoding, had little (or zero) effect on the model expressivity and trainability.

Typically IQR of 0.0000 indicates high stability, yet with sub-optimal performance, it signals a hard learning capacity ceiling.

Adding trainable layers without sequential data re-uploading fails to meaningfully increase expressivity.

Is this model (right) efficient and stable?

high training and testing MSE +
very low IQR at the optimum

Serial data reuploading created an expressive feature map + stability, which compensated for the loss of Hilbert space dimensionality within its ansatz.

Model	Type	Qubits	Params	Training R2 (IQR)	Testing R2 (IQR)	Training MSE (IQR)	Testing MSE (IQR)	Specs
xparallel	CF	1	69	0.9963 (0.0072)	0.9434 (0.0571)	0.0002 (0.0003)	0.0018 (0.0018)	q1 bl21 al1
xparallel	CF	3	81	**0.9996** (0.0002)	**0.9812** (0.0254)	**0.0000** (0.0000)	**0.0006** (0.0008)	q3 bl7 al1
xparallel	CF	5	75	0.9678 (0.0152)	0.9031 (0.0264)	0.0015 (0.0007)	0.0030 (0.0008)	q5 bl3 al1
xparallel	CF	7	84	0.8253 (0.1147)	0.7168 (0.1058)	0.0084 (0.0055)	0.0088 (0.0033)	q7 bl2 al1
xparallel	CF	9	81	0.6801 (0.2870)	0.5810 (0.3647)	0.0153 (0.0138)	0.0131 (0.0114)	q9 bl1 al1

CF Extended Parallel

Xqnn vs. OvLoad
forecasting models

FC Extended QNN
with Havlíček ZZ feature map

Model	Type	Qubits	Params	Training R2 (IQR)	Testing R2 (IQR)	Training MSE (IQR)	Testing MSE (IQR)	Specs
xqnn	FC	5	90	**0.9789** (0.0087)	0.5607 (0.2022)	**0.0008** (0.0003)	0.0154 (0.0071)	q5 in5 fm1 anz5
xqnn	FC	6	108	0.9353 (0.0536)	0.6676 (0.1303)	0.0025 (0.0021)	0.0116 (0.0046)	q6 in5 fm1 anz5
xqnn	FC	7	126	0.9772 (0.0177)	0.8358 (0.0707)	0.0009 (0.0007)	0.0057 (0.0025)	q7 in5 fm1 anz5
xqnn	FC	8	144	0.9366 (0.0567)	0.8113 (0.0727)	0.0025 (0.0022)	0.0066 (0.0025)	q8 in5 fm1 anz5
xqnn	FC	9	162	0.9655 (0.0278)	0.8437 (0.0799)	0.0014 (0.0011)	0.0055 (0.0028)	q9 in5 fm1 anz5
xqnn	FC	7	42	0.8946 (0.0458)	0.8410 (0.0581)	0.0042 (0.0018)	0.0056 (0.0020)	q7 in5 fm1 anz1
xqnn	FC	7	63	0.9579 (0.0180)	0.8535 (0.0766)	0.0017 (0.0007)	0.0051 (0.0027)	q7 in5 fm1 anz2
xqnn	FC	7	84	0.9629 (0.0205)	**0.8643** (0.0844)	0.0015 (0.0008)	**0.0047** (0.0030)	q7 in5 fm1 anz3
xqnn	FC	7	105	0.9665 (0.0318)	0.8456 (0.0994)	0.0013 (0.0013)	0.0054 (0.0035)	q7 in5 fm1 anz4
xqnn	FC	7	126	0.9772 (0.0177)	0.8358 (0.0707)	0.0009 (0.0007)	0.0057 (0.0025)	q7 in5 fm1 anz5
xqnn	FC	7	147	0.8928 (0.1596)	0.7310 (0.3260)	0.0042 (0.0063)	0.0094 (0.0114)	q7 in5 fm1 anz6
xqnn	FC	7	84	0.9629 (0.0205)	**0.8643** (0.0844)	0.0015 (0.0008)	**0.0047** (0.0030)	q7 in5 fm1 anz3
xqnn	FC	7	84	0.8505 (0.0314)	0.5829 (0.0886)	0.0059 (0.0012)	0.0146 (0.0031)	q7 in5 fm2 anz3
xqnn	FC	7	84	0.8705 (0.0210)	0.2569 (0.1413)	0.0051 (0.0008)	0.0260 (0.0049)	q7 in5 fm3 anz3

Is this model (left) efficient and stable?
stable training R2/MSE +
mediocre testing R2/MSE
unstable model, medium to high IQR

As time series windows are encoded in a highly entangling ZZ feature map, and the distinction between neighbouring values is inherently small, this scrambles the closely related states, creating a rugged cost landscape where training convergence is choppy and reliant on initialisation.

Addition of ancilla qubits, adds parameters and entanglements = excess entanglement, homogenising and randomising the feature space.

Is this model (right) efficient and stable?
high training and testing R2/MSE +
high stability - very low IQR

Qubits are overloading the representational function for the windows sub-sequence.

Qubits are reused, saving resource, thus providing structural regularisation, resulting in better performance + stability.

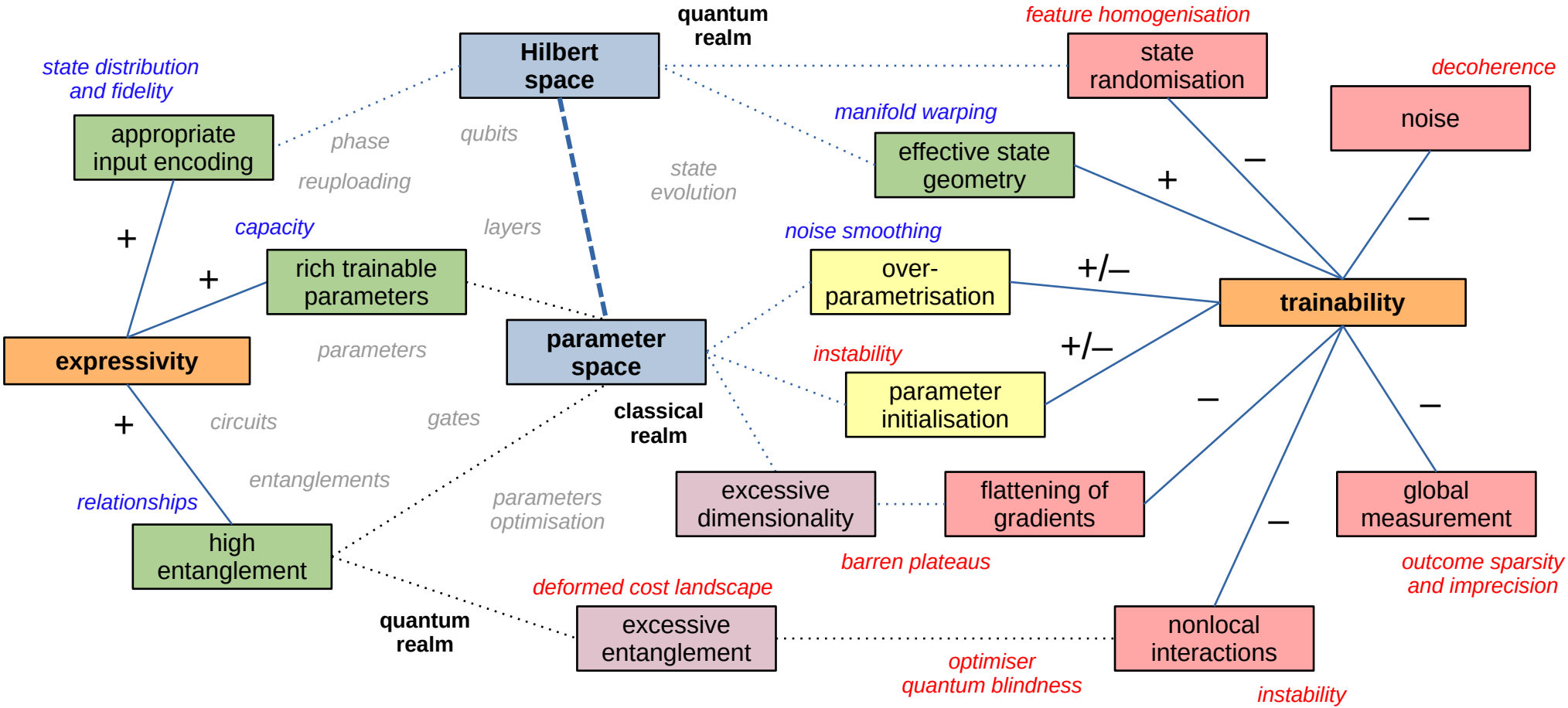
Model	Type	Qubits	Params	Training R2 (IQR)	Testing R2 (IQR)	Training MSE (IQR)	Testing MSE (IQR)	Specs
ovload	FC	1	92	0.2321 (0.0000)	0.1557 (0.0036)	0.0302 (0.0000)	0.0295 (0.0001)	q1 xl3 il5
ovload	FC	2	81	0.7856 (0.0000)	0.7697 (0.0013)	0.0084 (0.0000)	0.0081 (0.0000)	q2 xl2 il3
ovload	FC	3	88	0.8726 (0.0010)	0.8501 (0.0018)	0.0050 (0.0000)	0.0052 (0.0001)	q3 xl2 il2
ovload	FC	4	71	0.8848 (0.0021)	0.8663 (0.0021)	0.0045 (0.0001)	0.0047 (0.0001)	q4 xl1 il2
ovload	FC	4	119	0.8878 (0.0014)	0.8723 (0.0011)	0.0044 (0.0001)	0.0045 (0.0000)	q4 xl2 il2
ovload	FC	4	167	**0.8888** (0.0013)	**0.8734** (0.0018)	**0.0044** (0.0001)	**0.0044** (0.0001)	q4 xl3 il2

FC Overloading QNN
with structural regularisation

Expressivity vs. Trainability

lessons learnt for temporal quantum models

Where is Wally?
(stability)



Jacob L. Cybulski and Sebastian Zajac. “Quantum Modelling of Time Series: Expressivity vs. Trainability”,
20 August 2025, Preprint (Version 1) available at Research Square [https://doi.org/10.21203/rs.3.rs-7253212/v1].

from the literature and experiments

A quantum model should be like a duct tape, which is:

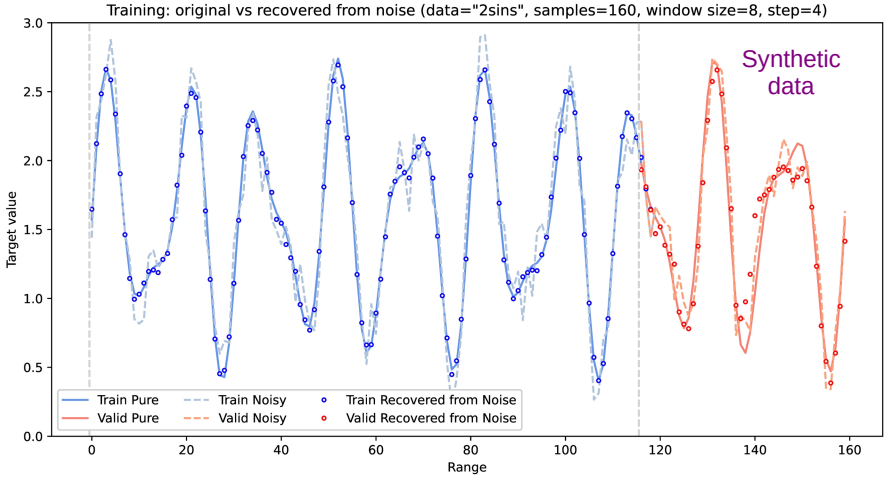
- *expressive, to stick to metal, plastic and wood*
- *trainable, to easily fold around any shape*
- *stable, to ensure it holds in different levels of humidity and temperature for a very long time*

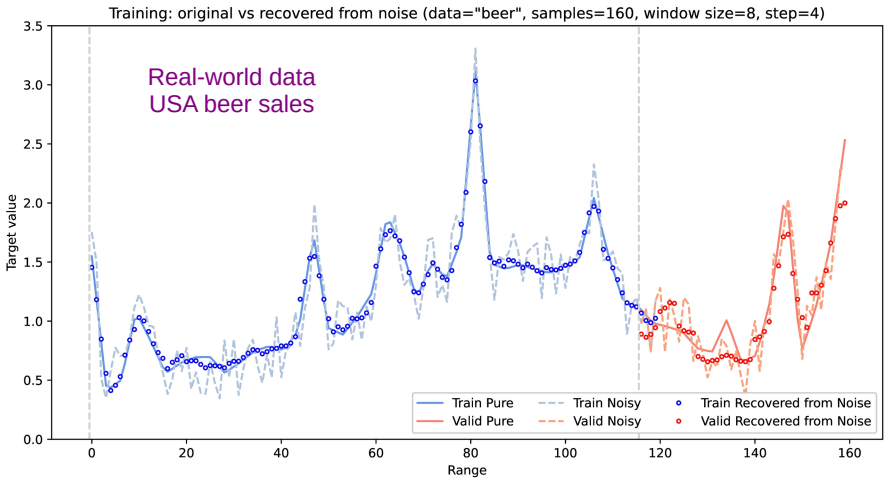
The Research Journey

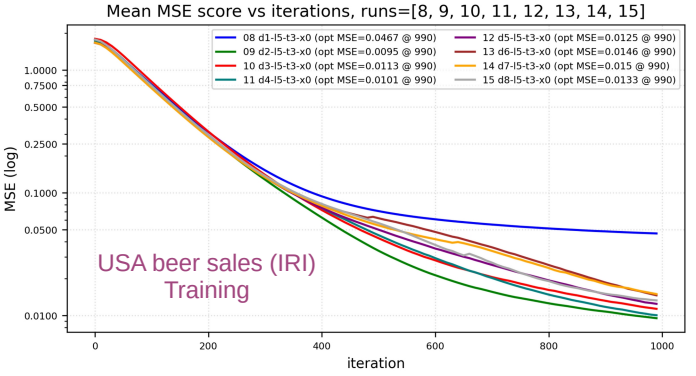
From models to metrics -
when to continue or stop the pursuit of a perfect model?

The concepts of expressivity, trainability and stability have been defined in a multi-faceted way and their presence in quantum models detected objectively using a variety of metrics

In search of trainability (informally)







Denoising QAEs

Are QAE models **expressive** enough to remove noise from complex TS data?

Are they **trainable** on real data?

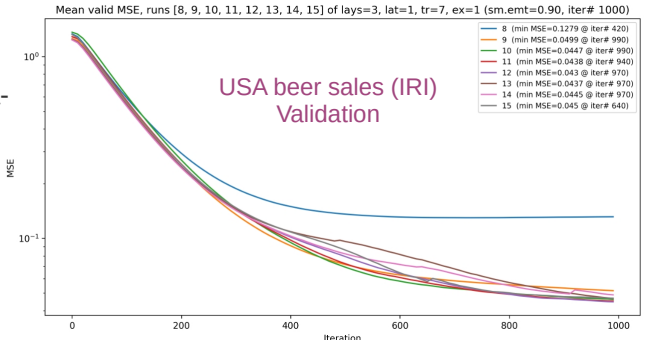
Are they flexible enough to be **stable** across training / test data sets, and across domains?

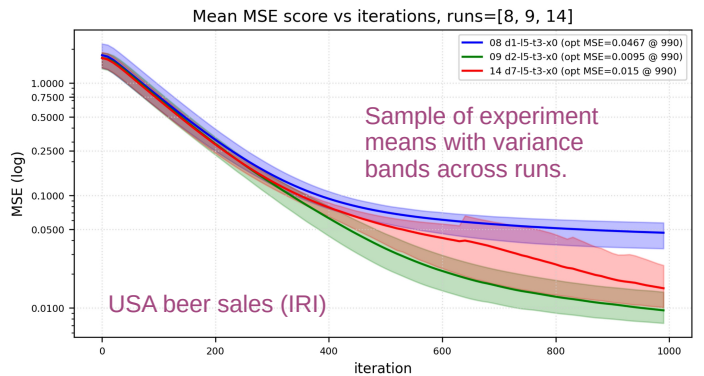
Initially the QAEs were developed using simple synthetic time series.

This was followed by their adaptation in structure, size and optimisation to work with real-life data.

Success was judged with mean performance and variance.

This research raised the question of how to determine the model capacity to learn.





Different QML platforms were investigated, with CPU and GPU support.

Qiskit where models converged within 1000 epochs.

PennyLane + PyTorch where models learnt quickly and had capacity to improve beyond the 1000 epochs.

Jacob L. Cybulski and Sebastian Zając (2024): "Design Considerations for Denoising Quantum Time-Series Autoencoder." In Proc. of 24th Int. Conf. on Comp. Sci. (ICCS), Malaga, Spain.

In search of expressiveness (informally)

Can model *capacity to learn* be used to assess effectiveness of barren plateau mitigation strategy?

global effective dimension (GED)

Combines Vapnik and Chervonenkis Effective Dimension and Fisher Information Matrix. The model *expressivity* was represented as a static, probabilistic measure of the complexity of the parameter space geometry (via gradients).

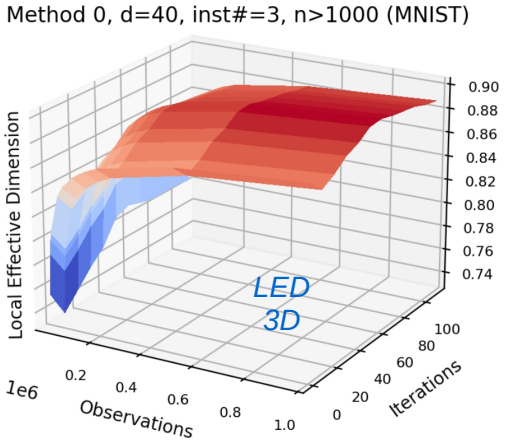
local effective dimension (LED)

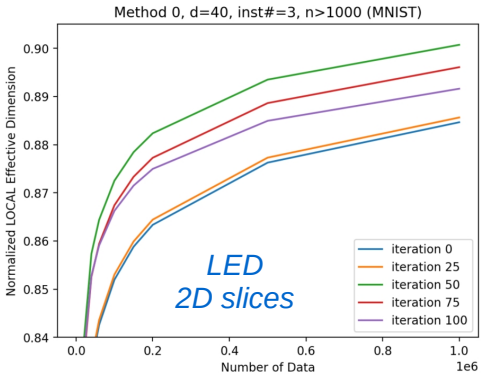
The model *trainability* was represented by the dynamic geometric measure of the model complexity in training, similar to GED but affected by data distribution and optimisation algorithm.
The model *stability* was assessed using the variance of instance LED.

Cybulski, J.L., Nguyen, T., 2023. Impact of barren plateaus countermeasures on the quantum neural network capacity to learn. Quantum Inf Process 22, 442.

Abbas, A., Sutter, D., Figalli, A., Woerner, S., 2021a. Effective dimension of machine learning models, arXiv:2112.04807.

Abbas, A., Sutter, D., Zoufal, C., Lucchi, A., Figalli, A., Woerner, S., 2021b. The power of quantum neural networks. Nat Comput Sci 1, 403–409.



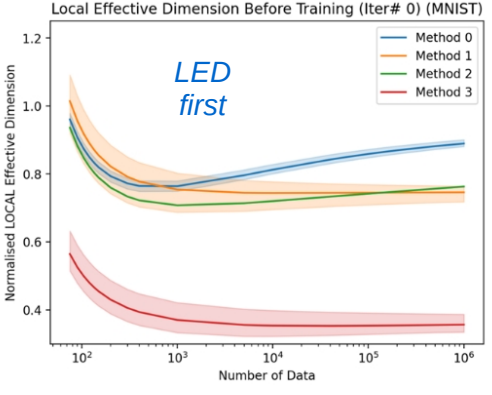


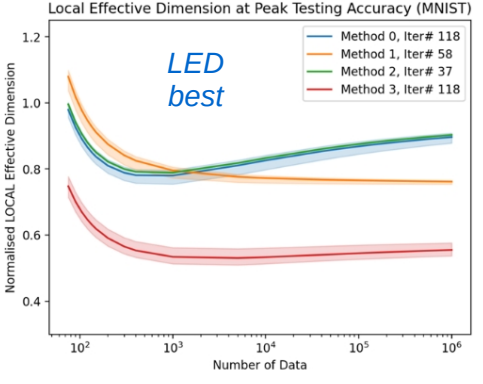
Each BP method was tested using a QNN classifier.
Each classifier had 10 instances.
Each instance was trained on Iris and MNIST data.

GED and LED are calculated classically.
GED sets the LED upper bound

The LED results were depicted in 3D.
To simplify LED 3D visualisation we 2D plot only the selected data slices, which were depicted as variance.

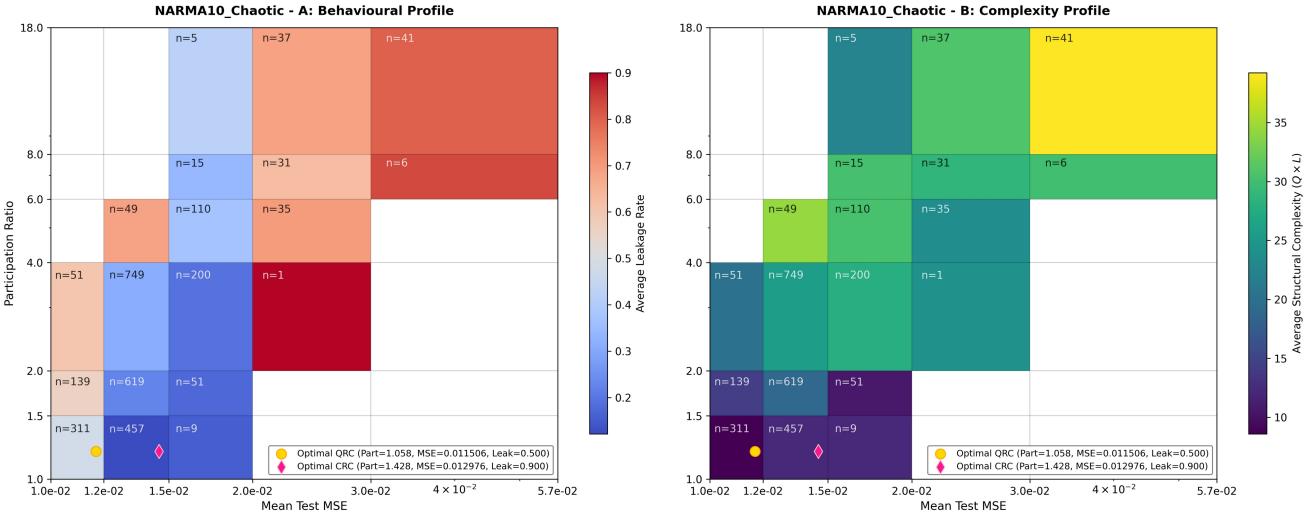
LED evolves with training
LED scales with test scores
LED variance reduces with training

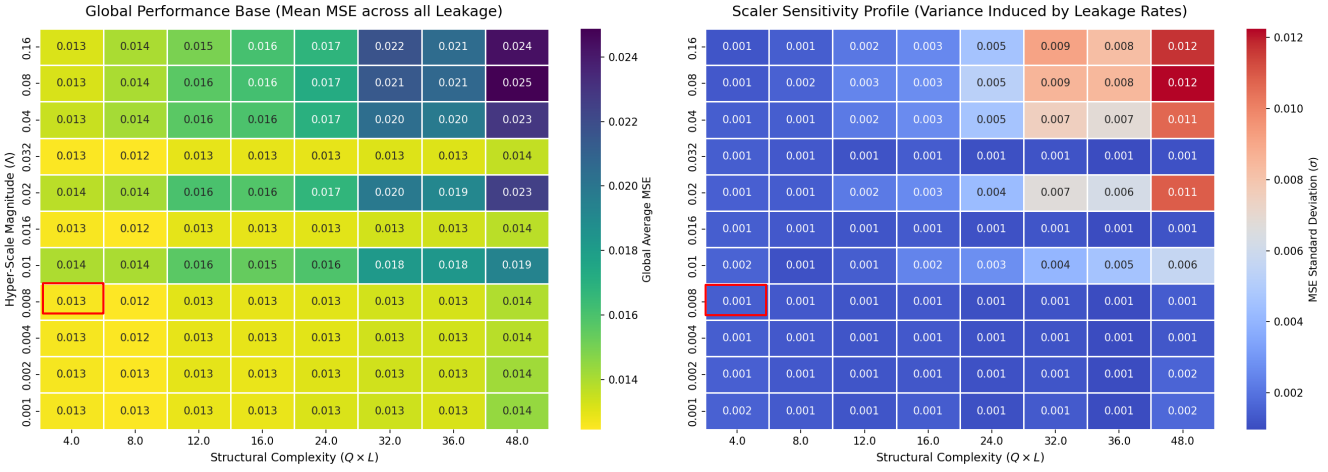




In search of stability (informally)

Jacob L. Cybulski, Stanisław Władyka and Sebastian Zając (2026):
"Memory duality for quantum reservoir computing." In preparation.





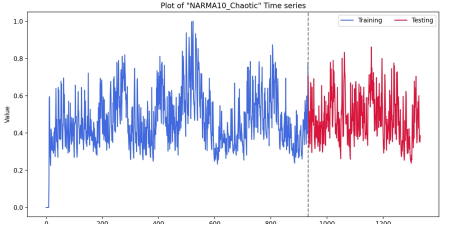
Quantum Reservoir Models

The study of the performance and stability of QRC models as applied to chaotic time series (Mackey-Glass, NARMA, ARMA, etc.).

In this research QRC combined the reservoir memory with the sliding windows, which were combined and mapped into high-dimensional, warped Hilbert space manifold.

The model *expressivity* was represented as information participation ratio (from PCA) reaching the classical readout. Its *trainability* as cost / scores curves. The *stability* measured as σ(MSE) within the model structure (qubits x layers).

QRC produced a superior performance, with small model complexity and stable MSE across all its variants.

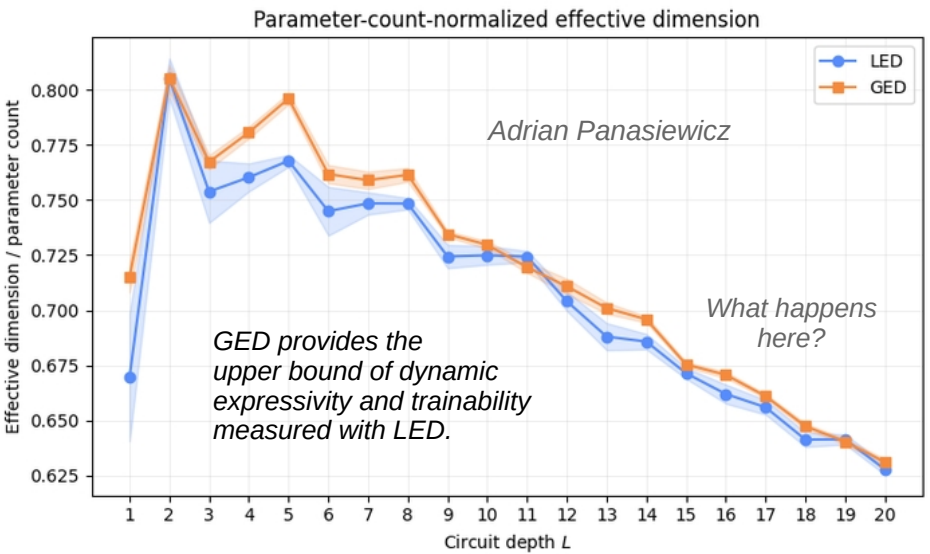


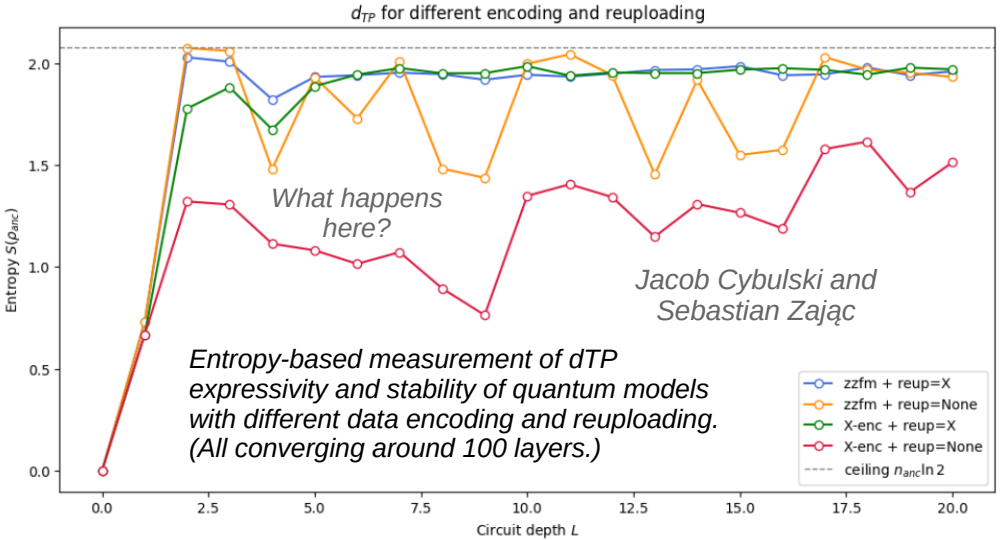
$$\mathbf{x}^{cl}(t) = (1-\gamma)\mathbf{x}^{cl}(t-1) + \gamma\mathbf{u}(t)$$

Current work
in metrics for quantum models

We are further developing Effective Dimension metrics (global and local) to assess various quantum model design issues, such as the type of data encoding, data reuploading, the circuit width (qubits number) and depth (layers number) (below).

The focus is on the geometry of the parameter space, which could be measured without model training.



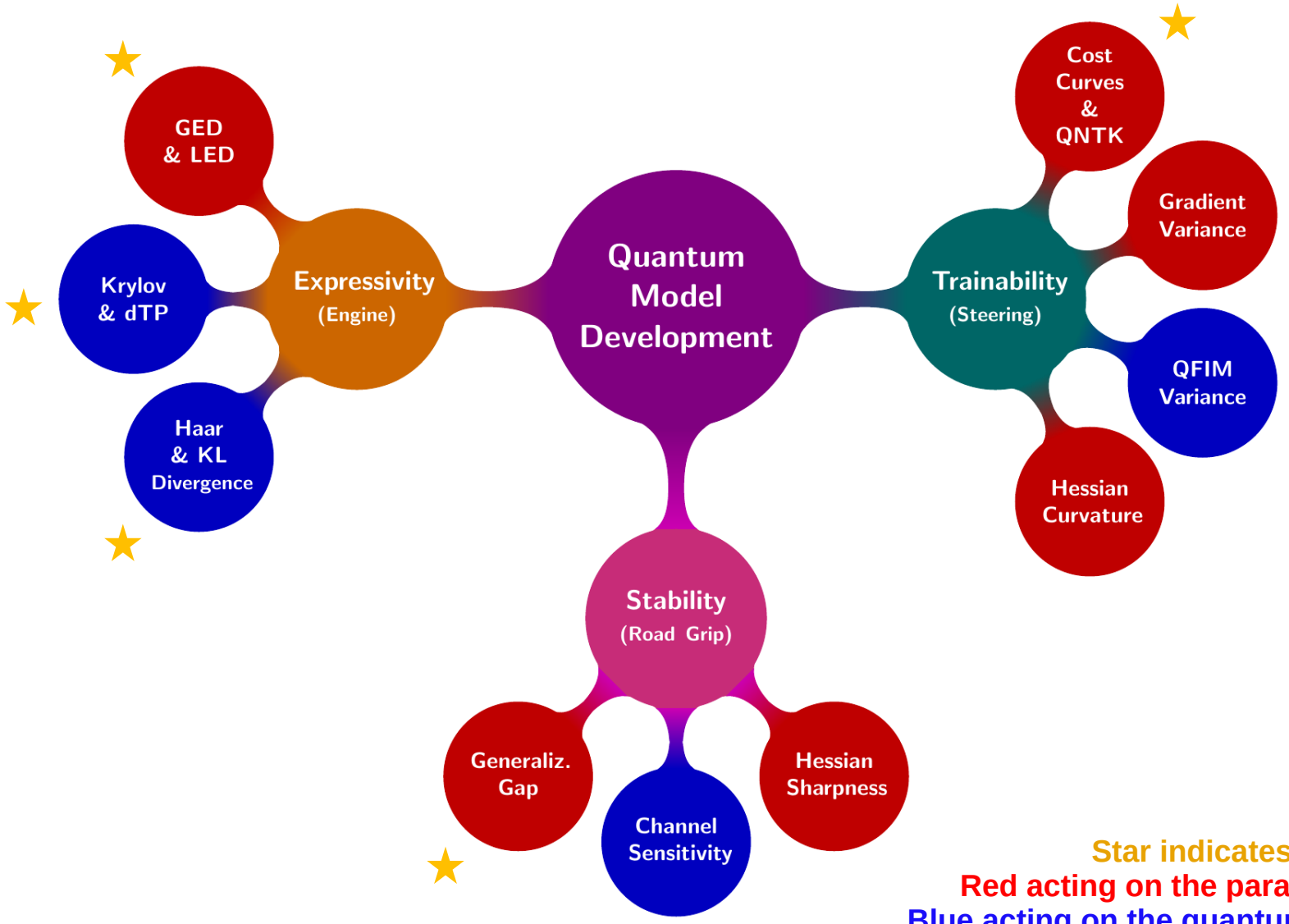


We also assess the geometry of the quantum state space manifolds for their capacity to accommodate model transitions through the Hilbert space (above).

We propose a Trajectory Participation Dimension (dTP) metric, inspired by Krylov expressivity - a local, state-space diagnostic for parameterised quantum circuits. This metric can be applied to untrained (and trained) models, to assess their expressivity and stability even before optimisation.
As dTP cannot access the Hilbert space directly,
thus at this stage we calculate it using a quantum simulator.

Other work includes experiments with Haar / KL expressivity and QNTK spectrum analysis.

Expressivity, Trainability and Stability Metrics



Star indicates current work
Red acting on the parameters space
Blue acting on the quantum state space

Summary and the key takeaways

- **Expressivity (The Engine):** Raw Hilbert space capacity is useless if highly correlated time series data gets scrambled. Serial data re-uploading is non-negotiable for combating feature homogenisation and maintaining state discernibility.
- **Trainability (The Steering):** Uncontrolled expansion of circuit width and depth introduces excessive entanglements that warp the cost landscape. Narrow, entanglement-controlled circuits are far easier for the classical optimizer to navigate without stalling on barren plateaus.
- **Stability (The Road Grip):** High expressivity and trainability often mask a vulnerability to overfitting and wild variances in testing. Qubit overloading (reusing qubits for sliding windows) acts as a structural regularisation, locking in stability and ensuring consistent, reliable performance on unseen data.
- **The diagnostic toolkit:** Theoretical capacity means nothing if the optimizer gets lost. Metrics like GED, LED, and dTP provide the radar needed to evaluate the expressivity, trainability, and stability of a circuit before running a single epoch.
- **Where do we go from here?** The QIFT program...

Sebastian Zając and Jacob L. Cybulski (2026), "Quantum Data Analysis: From Data Points to Discrete Quantum Information Fields." In preparation.

Sebastian Zając and Jacob L. Cybulski (2026), "Fermionic Quantum Information Field Theory: Anti-Redundancy Constraints, the Informational Pauli Exclusion Principle, and a New Class of Structural Anomaly." In preparation.

Thank you!
Any questions?

Acknowledgements:

We would like to thank our colleagues Bartosz Dziewit and Tomasz Kulpa, as well as the current and past interns and research scholars who contributed to this work, i.e. Adrian Panasiewicz, Miron Hunia, Marysia Nazarczuk, and Marcin Klaczak (AIntern 2026), Stanisław Władyka, Jakub Zwoniarski, and Artur Strąg (AIntern 2025), Thanh Nguyen (Deakin Summer Scholarship 2022), as well as Paul Gora, CEO of Fundacja Quantum AI, who facilitated some of the internships.

Mathematical model definitions and more results in the Appendix...

Curve-fitting models

$$|\psi(x,\boldsymbol{\theta})\rangle = U_{rot}(\boldsymbol{\theta}_{n+1}) \cdot \left(\prod_{j=1}^{n} [R_X(x) \cdot U_{rot}(\boldsymbol{\theta}_j)] \right) \cdot |0\rangle \,,$$

$$y = \langle 0|^{\otimes q} U^\dagger M U \, |0\rangle^{\otimes q} \,,$$

$$U(\vec{x},\vec{\theta}) = U_{\text{ansatz}}(\vec{\theta}) \cdot \mathcal{U}_{\text{FM}}(\vec{x}),$$

$$U_{\text{ent}}(\boldsymbol{\theta}_j) = \prod_{i=1}^{q} \text{CNOT}_{i,\,(i \bmod q)+1} \cdot U^q(\boldsymbol{\theta}_j)$$

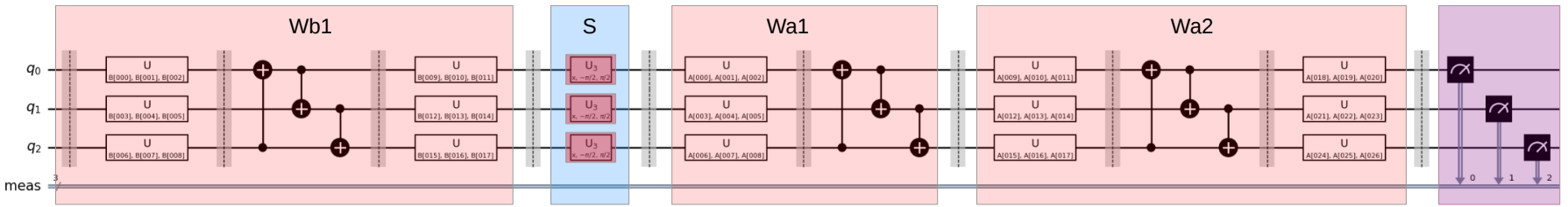
$$U^q(\boldsymbol{\theta}_j) = \bigotimes_{i=1}^{q} U_{rot}^{(i)}(\boldsymbol{\theta}_j^{(i)})$$

$$R_X^q(x) = \bigotimes_{j=1}^{q} R_X^{(j)}(x).$$

CF Serial

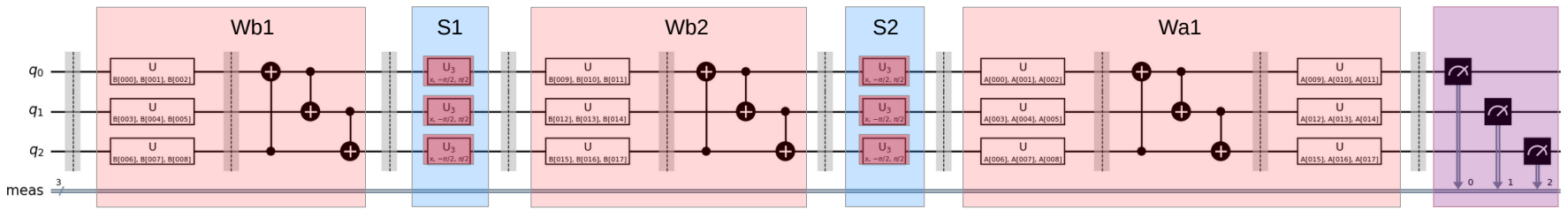
$$|\psi(x,\boldsymbol{\theta})\rangle = \left(U^q(\boldsymbol{\theta}_{n+1}) \prod_{j=m+2}^{n} [U_{\text{ent}}(\boldsymbol{\theta}_j)] \right) R_X^q(x) \left(U^q(\boldsymbol{\theta}_{m+1}) \prod_{k=1}^{m} [U_{\text{ent}}(\boldsymbol{\theta}_k)] \right) |0\rangle^{\otimes q}$$

CF Parallel



$$|\psi(x,\boldsymbol{\theta})\rangle = \left(U^q(\boldsymbol{\theta}_{m+1}) \prod_{k=L+1}^{m} U_{\text{ent}}(\boldsymbol{\theta}_k) \right) \cdot \left(\prod_{l=1}^{L} [R_X^q(x) \cdot U_{\text{ent}}(\boldsymbol{\theta}_l)] \right) |0\rangle^{\otimes q}$$

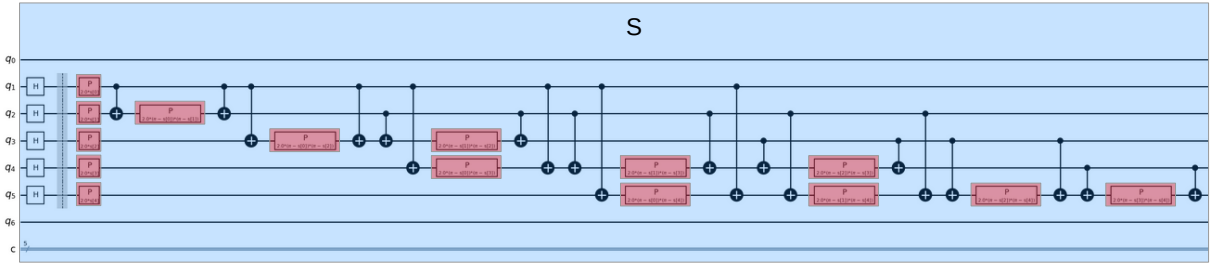
CF Extended Parallel



$$U_{\Phi(\mathbf{x})} = \exp\left(i \sum_{S \subseteq [n]} \phi_S(\mathbf{x}) \prod_{i \in S} Z_i\right)$$

$$\phi_{\{i\}}(\hat{\mathbf{x}}) = x_i$$

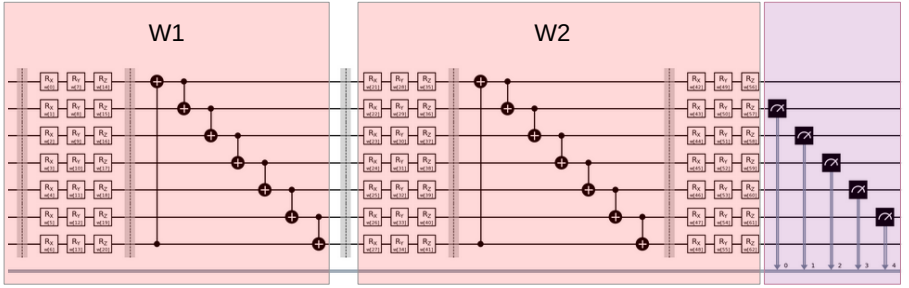
$$\phi_{\{i,j\}}(\hat{\mathbf{x}}) = (\pi - x_i)(\pi - x_j)$$



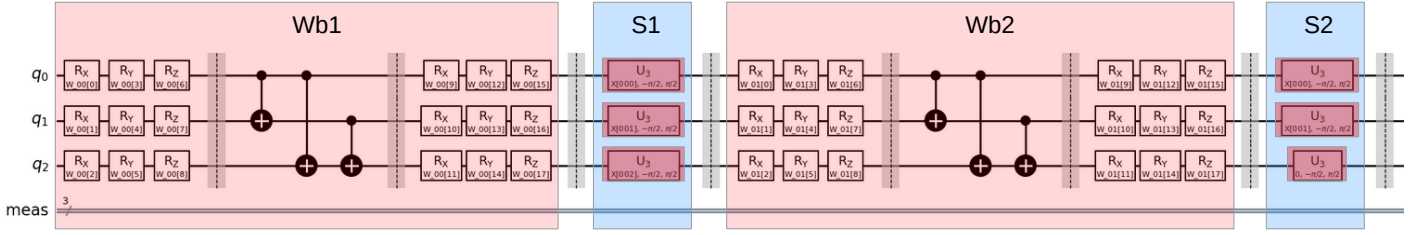
$$\mathcal{U}_{FM}(\mathbf{x}) = U_{\Phi(\mathbf{x})} \cdot H^{\otimes n}$$

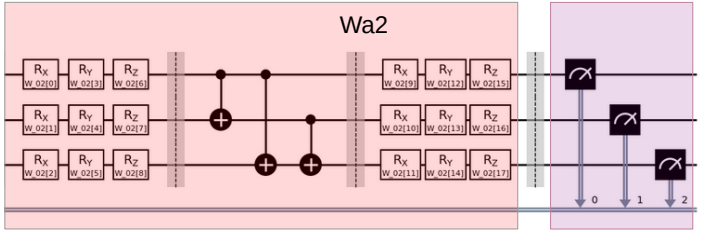
$$U_{AN}(\boldsymbol{\theta}) = \prod_{k=1}^{L_{AN}} U_{\text{ent}}(\boldsymbol{\theta}_k)$$

$$|\psi(\mathbf{x}, \boldsymbol{\theta})\rangle = U_{AN}(\boldsymbol{\theta}) \cdot \mathcal{U}_{FM}(\mathbf{x}) |0\rangle^{\otimes q}$$



FC Extended QNN
with Havlíček ZZ feature map





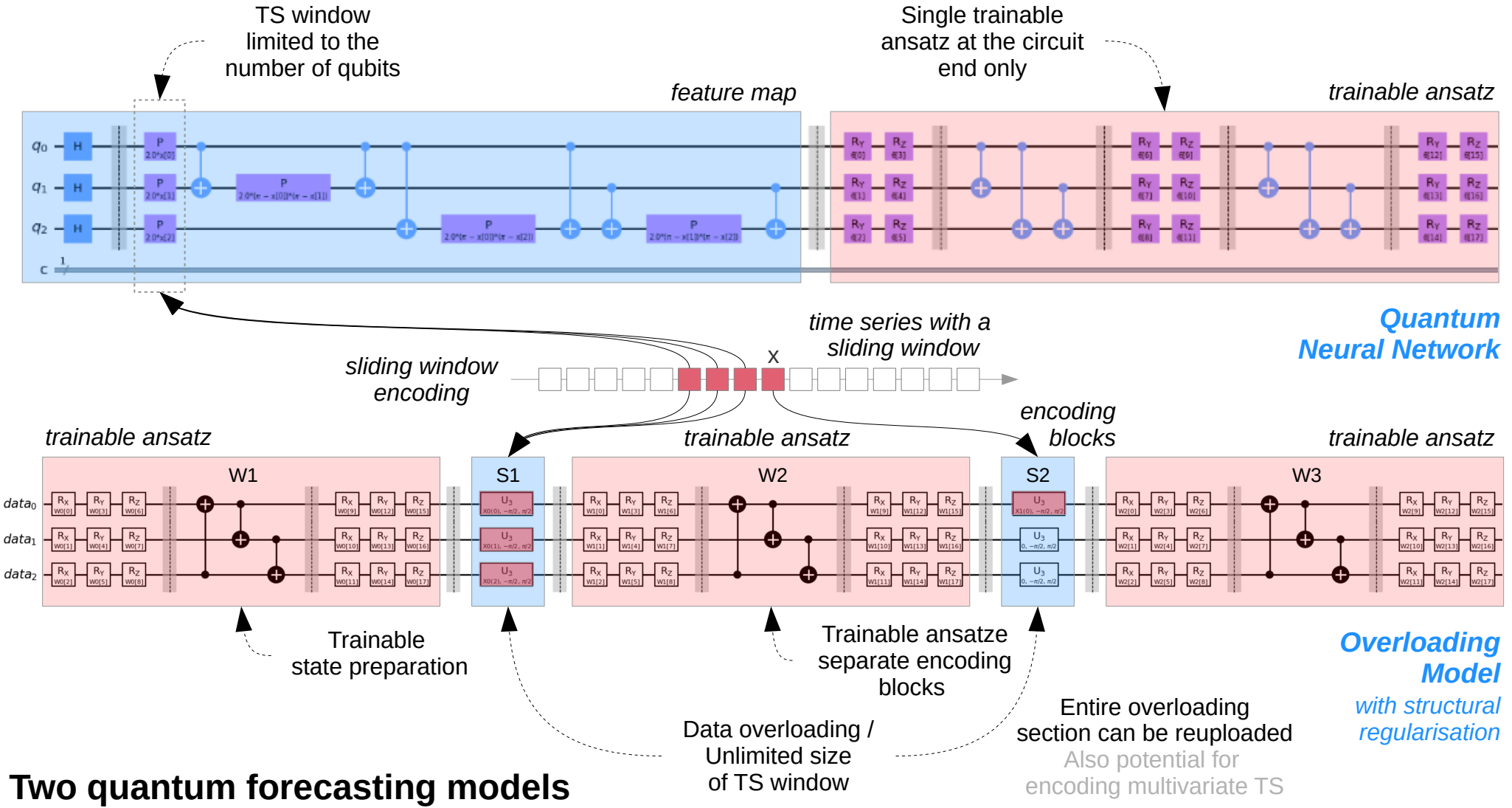
FC Overloading QNN
with structural regularisation

$$W_{\text{ent}}^{l}(\hat{\boldsymbol{\theta}}) = U^{q}(\boldsymbol{\theta}_{l+1}) \cdot U_{\text{ent}}(\boldsymbol{\theta}_l)$$

$$\mathcal{R}_X^q(\hat{\mathbf{x}}) = \prod_{k=0}^{\lceil n/q \rceil - 1} \left(\bigotimes_{j=1}^{q} R_X^{(j)}(x_{kq+j}) \right)$$

$$\left|\psi(\hat{\mathbf{x}}, \hat{\boldsymbol{\theta}})\right\rangle = W_{\text{ent}}^{L+1}(\hat{\boldsymbol{\theta}}) \cdot \left(\prod_{k=1}^{L} \left[\mathcal{R}_X^q(\hat{\mathbf{x}}) \cdot W_{\text{ent}}^{k}(\hat{\boldsymbol{\theta}}) \right] \right) |0\rangle^{\otimes q}$$

Forecasting models



Two quantum forecasting models

Comparison of different QTSA architectures

Scoring results for the best model variants with median instance performance

The task of curve-fitting models was simple, they learnt a function from data, to map x coordinates to y values.
Note the amazing performance of the 1 qubit model!
The task of forecasting models was harder, as they learnt to predict the future value from the preceding context.

Data was a trigonometric function. As all CF models used angle-encoding, they acted as truncated quantum Fourier transform facilitated by Pauli rotations as harmonics and frequencies generated by eigenvalues and reuploading.
Not surprising on R2, CFs performed better than FCs.

	Model	Type	Q#	P#	Training R2 (IQR)	Testing R2 (IQR)	Training MSE (IQR)	Testing MSE (IQR)	Specs
Quantum Curve-fitting	serial	CF	1	66	0.9926 (0.0060)	0.9122 (0.1617)	0.0004 (0.0003)	0.0027 (0.0050)	q1 l21
	serial	CF	1	84	0.9947 (0.0053)	0.8912 (0.1982)	0.0003 (0.0003)	0.0034 (0.0062)	q1 l27
	parallel	CF	5	120	0.8620 (0.0000)	0.6939 (0.0283)	0.0066 (0.0000)	0.0095 (0.0009)	q5 bl3 al3
	parallel	CF	5	90	0.8620 (0.0001)	0.7109 (0.0374)	0.0066 (0.0000)	0.0090 (0.0012)	q5 bl1 al3
	xparallel	CF	3	81	0.9996 (0.0002)	**0.9812** (0.0254)	0.0000 (0.0000)	**0.0006** (0.0008)	q3 bl7 al1
Quantum Forecasting	xqnn	FC	5	90	0.9789 (0.0087)	0.5607 (0.2022)	0.0008 (0.0003)	0.0154 (0.0071)	q5 in5 fm1 anz5
	xqnn	FC	7	84	0.9629 (0.0205)	0.8643 (0.0844)	0.0015 (0.0008)	0.0047 (0.0030)	q7 in5 fm1 anz3
	ovload	FC	4	167	0.8888 (0.0013)	0.8734 (0.0018)	0.0044 (0.0001)	0.0044 (0.0001)	q4 xl3 il2
Classical Forecasting	cnn tiny	FC	0	106	1.0000 (0.0021)	0.9684 (0.0420)	0.0000 (0.0001)	0.0011 (0.0015)	hl008 005 003
	cnn small	FC	0	375	1.0000 (0.0000)	0.9771 (0.0008)	0.0000 (0.0000)	0.0008 (0.0000)	hl010 015 010
	cnn medium	FC	0	5950	**1.0000** (0.0000)	0.9796 (0.0010)	**0.0000** (0.0000)	0.0007 (0.0000)	hl050 080 020

Note: Q# = Qubit Number, P# = Params Number, CF = Curve-Fitting, FC = Forecasting.
bold = best overall, red / blue = best training / testing for their model architecture.

The "Large" Cnn model with 20,800 parameters was excluded from this comparison, as the model had over 100 times more parameters than the largest quantum model.

Jacob L. Cybulski (Jul 2026): Code, Results and Supplement for the QUANCOM'2026 paper on Quantum Modelling of Time Series - Expressivity vs. Trainability, https://github.com/ironfrown/ts_expressivity_v1.

Measuring *expressivity* of quantum models

Can we formally and rigorously introduce expressivity?

Expressivity
is the model's capacity to represent complex data within the quantum space.

Metric	Primary Focus	Scope	What it tells us
Grid-Testing (impractical)	Parameter Space (Boundary)	Local (Data-dependent)	What is the performance boundary for given data and each combination of weight parameters?
Haar / KL Divergence	States Uniformity (Fidelities)	Global	Can this circuit reach anywhere in the Hilbert space?
QNTK	Gradient Space (Geometry)	Local	Will gradient descent actually find a solution?
Global Effective Dimension	Parameter Space (Information)	Global	What is the upper bound of the model's learning capacity?
Local Effective Dimension	Parameter Space (Information)	Local (Data-dependent)	Is this model the right size for this specific dataset?
d_TP Trajectory Participation Dimension	Trajectory Dynamics (Geometry+Entropy)	Local	How much of Hilbert state space a circuit's trajectory actually visits?
Krylov Expressivity	Subspace Dynamics (Geometry)	Global	What is the boundary of states this circuit can explore?

Blue *indicates approaches currently investigated by our team*

Measuring *trainability* of quantum models

Can we formally and rigorously introduce trainability?

Trainability
is the model's navigability—how easily an optimiser can find the best configuration.

Metric	Primary Focus	Scope	What it tells us
			Classical Optimisation
Cost and Score Curves	Empirical Monitoring of Training Process	Local (Data-dependent)	What is the real-time health of the training process? At what point the model overfits training data?
Gradient Variance (Scaling)	Gradient Landscape (Statistical)	Global	Will the gradient signal vanish exponentially as the system scales (Barren Plateaus)?
QNTK (Trace / Spectrum)	Gradient Space (Geometry)	Local (Data-dependent)	Is the local optimization space well-conditioned enough to guarantee convergence?
Hessian Spectrum (Curvature)	Parameter Space (Geometry)	Local	Are there sharp ravines or flat valleys that will stall or trap the classical optimizer?
Dynamical Lie Algebra (DLA)	Operator Algebra (Symmetry)	Global	Is the model structurally constrained enough to avoid expressivity-induced barren plateaus?
Empirical Gradient Norms	Optimization Trajectory (Empirical)	Local (Run-dependent)	Is the specific training configuration actively suffering from vanishing or exploding gradients?
			Quantum Natural Gradient Optimisation
QFIM Spectrum & Rank	Fubini-Study Metric (Geometry)	Local	Is the state-space geometry stable enough to be inverted for a natural gradient step?
QFIM Variance Scaling	Fubini-Study Metric (Statistical)	Global	Will the natural gradient advantage vanish exponentially as the system scales (Natural Barren Plateaus)?

Blue *indicates approaches currently investigated by our team*

Measuring *stability* of quantum models

Can we formally and rigorously introduce stability?

Stability
is the model's consistency in producing reliable results, despite minor changes to its data or operating environment.

Metric	Primary Focus	Scope	What it tells us
Hessian Eigenspectrum (Sharpness)	Parameter Space (Geometry)	Local	How sensitive is the trained model to small parameter shifts or hardware control errors?
Lipschitz Continuity Bound	Input Space (Geometry)	Global	Is the quantum model mathematically robust against adversarial noise or small shifts in the classical input data?
Depolarizing Channel Sensitivity	State Space (Physical)	Global	How rapidly does the model's output fidelity degrade in the presence of physical hardware noise?
Generalization Gap	Statistical Learning (Information)	Global (Data-dependent)	Does the model maintain its performance on unseen population data, or has its expressivity caused it to overfit to training data or noise?
QNTK Kernel Alignment	Gradient Space (Alignment)	Local (Data-dependent)	Does the model's inherent geometric bias align stably with the target function across different subsets of data?

Blue *indicates approaches currently investigated by our team*