The aims of this session:

To explore approaches to reservoir computing and introduce a dual memory quantum reservoir model

Introduction
Reservoir Computing: CRC and QRC
DM-QRC information flow
QRC parameterisation
Formal definition of QRC
Types of RCs and QRCs
DM-QRC data encoding issues
DM-QRC initialisation sensitivity
Working with chaotic data sets
Measuring DM-QRC and its performance
DM-QRC case study
Summary of results
Summary and reflection
Selected readings

Quantum Reservoir Computing

and the memory duality

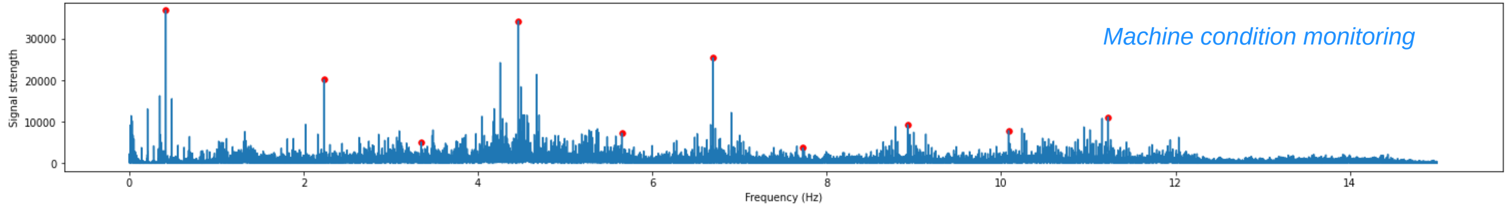
Jacob L. Cybulski
Enquanted, Melbourne, Australia

Stanisław Władyka
AIntern Program, Quantum AI Foundation, Poland

Sebastian Zając
Military University of Technology, Warsaw, Poland

11 July 2026, Warsaw Quantum Computing Group / QAIF, Warsaw, Poland

Problem



Machine condition monitoring

EEG analysis

Reservoir computing (RC) demonstrated highly efficient analysis of time series and signals. This project explores the potential of QML to further improve RC algorithms.

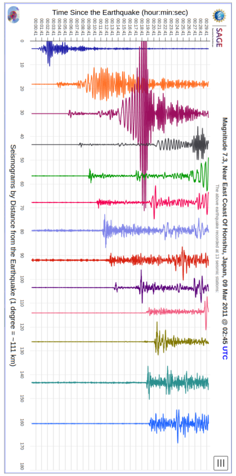
Classical RC was successfully deployed in prediction and forecasting, classification of temporal patterns, noise reduction and anomaly detection in signals, modelling of chaotic systems.

Potential QRC applications: machine condition monitoring, astronomical observations, marketing, earthquake prediction, EEG or ECG analysis, etc.

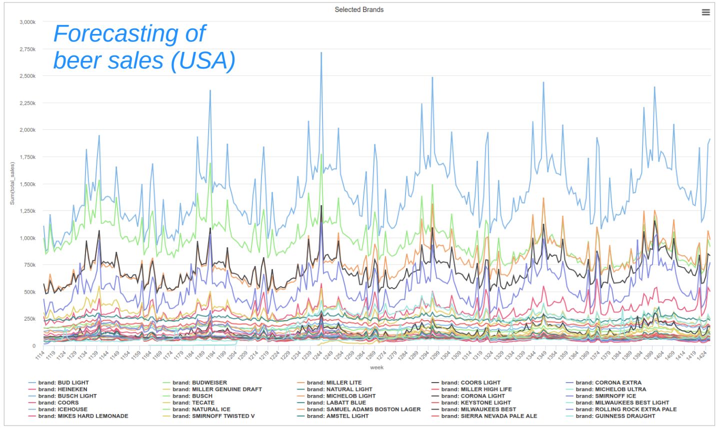
all data are temporal

all suffer from noise and anomalies

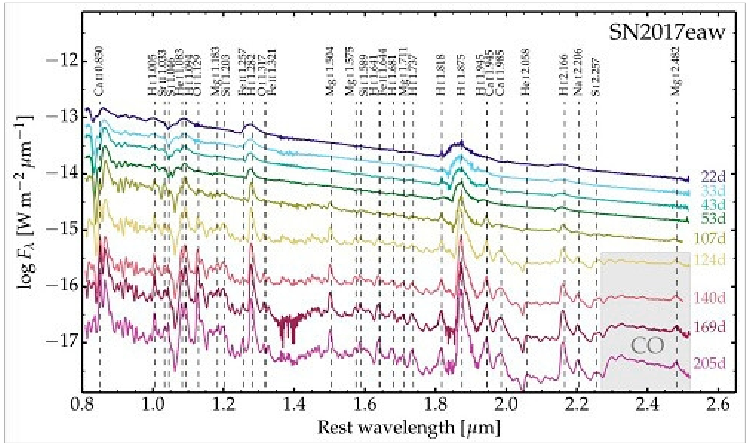
all could potentially lead to catastrophic failures



Earthquakes



Forecasting of beer sales (USA)



Astronomical observations

References:

Gearbox and Vibration Analysis ML, 2023, Nakkeeran, Kaggle.com

Gemini/GNIRS spectra, 2017, NOIRLab, Wikimedia.

Bronnenberg, B.J., Kruger, M.W., Mela, C.F., 2008. Database Paper — The IRI Marketing Data Set. Marketing Science 27, 745–748. https://doi.org/10.1287/mksc.1080.0450

Earthquake, Mag 7.3, East Coast of Honshu, Japan, 2011, The Global Seismogram Viewer, http://ds.iris.edu/gsv

EEG of brain and heart action, 2012, Otoomuch, Wikimedia.

Jacob L. Cybulski and Sebastian Zając (2024): "Design Considerations for Denoising Quantum Time-Series Autoencoder." In Proc. of 24th International Conference on Computational Science (ICCS), July 2-4, 2024, Malaga, Spain.

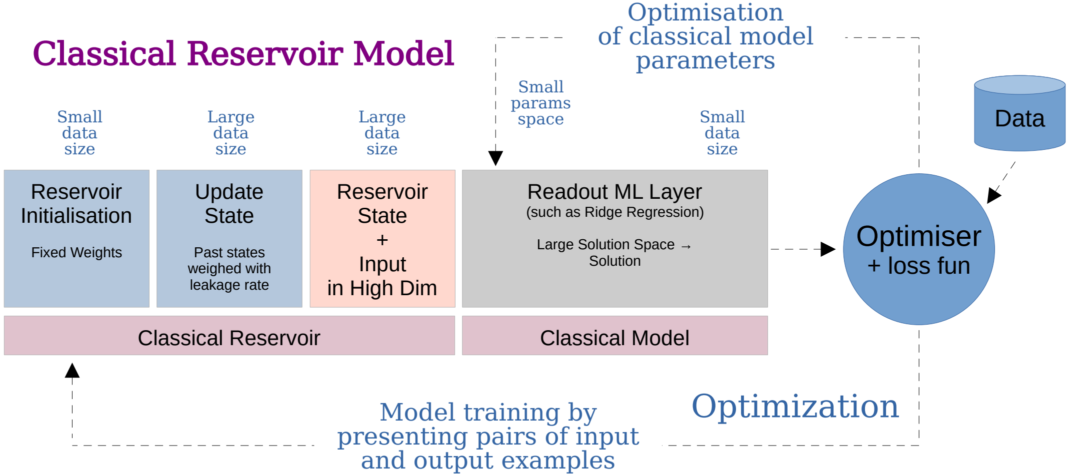
Reservoir computing

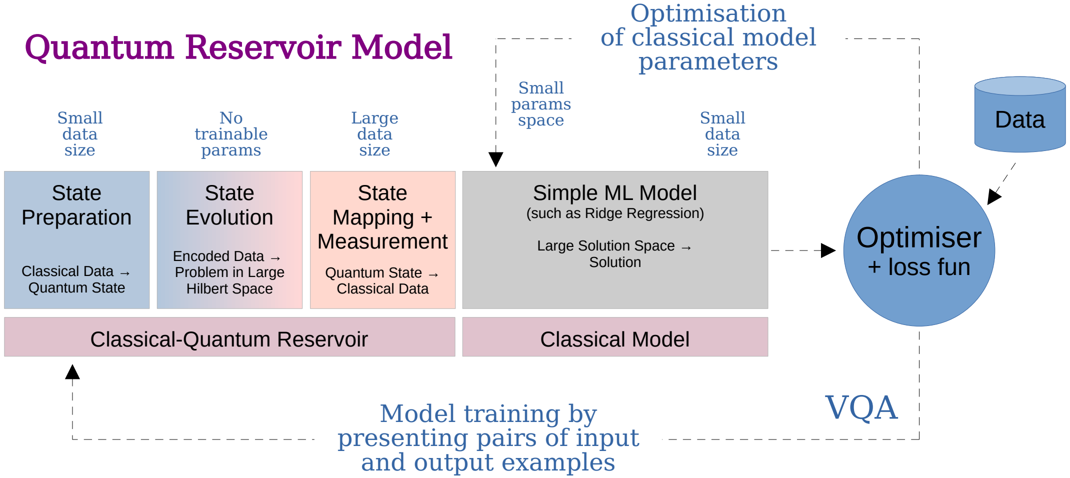
for series and signals

There are many types of classical and quantum reservoir models, e.g. echo-state networks with leaky integrator, feedback-driven, liquid state machine, next generation, time-delay, dissipative, physical CRCs/QRCs.

Classical RCs (CRC) are derived from recurrent neural networks. They are especially useful when working with *temporal data*. Reservoir computing utilises a *reservoir* – a large sparse neural network of randomly initialised and fixed weights, which transform input into a *higher-dimensional space*. In high-dimensional space, *data can be easily separated* (classified) by using a simple linear model (readout layer), such a ridge regression.

The reservoir should be able to *echo* or *retain information* about past inputs for a short period, making it suitable for processing sequential data. However, the influence of past input on the reservoir state should fade away over time – *memory leakage*.





Hybrid Quantum-Classical QRC

Hybrid quantum reservoir models consist of two parts:

- ***quantum reservoir model*** (echo and fading / leakage)
- ***classical readout model*** (e.g. ridge regression)

QRC takes a single TS value, however, it could take more.

QRC weights are sparse, random and fixed (no training).

QRC aim of its quantum part is to increase data separability by projection into non-linear high dimensional state space (kernel).

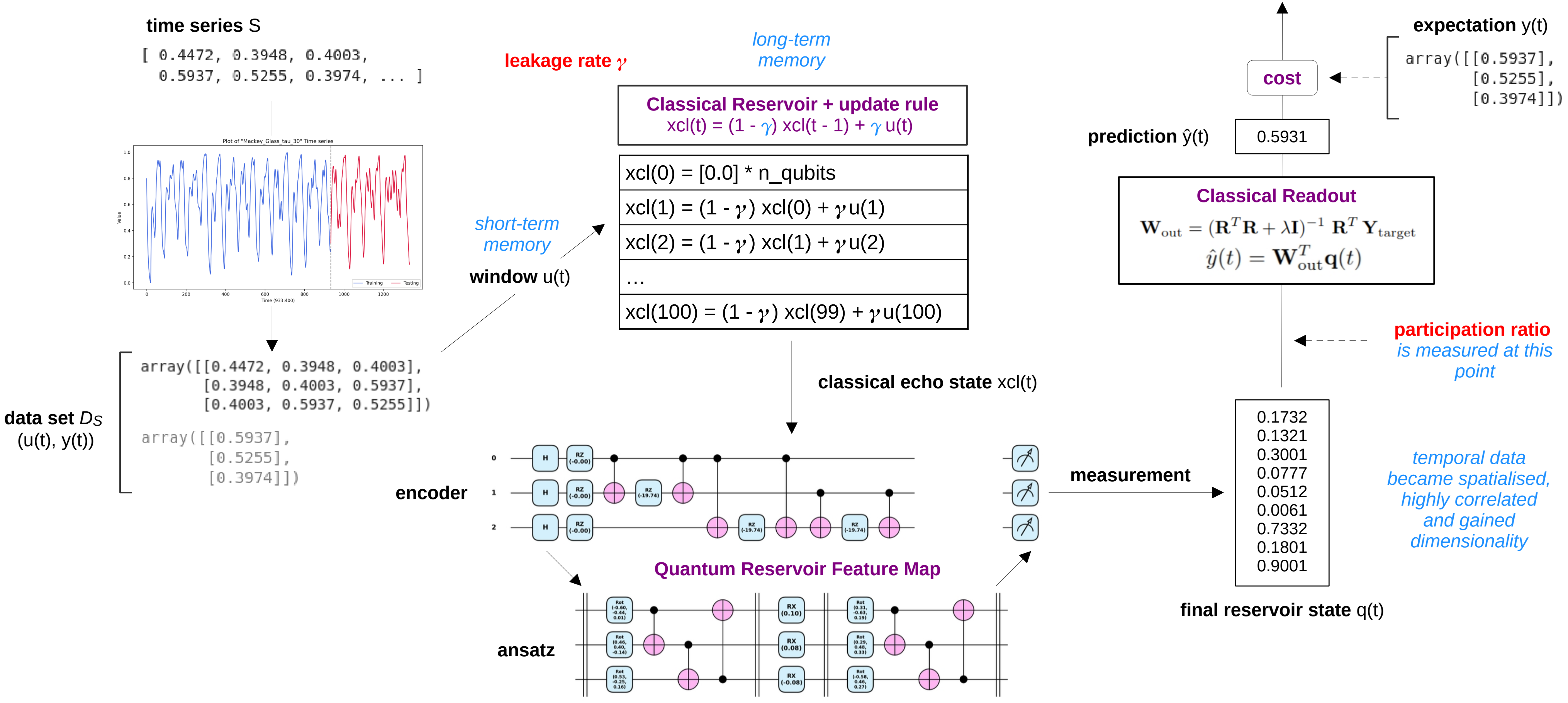
QRC classical readout is very simple and easy to train.

As there is no training of a quantum component, hybrid reservoir models can be trained very efficiently on large data sets.

Dual Memory QRC

and its information flow

*Let us start with a discussion of our Quantum Reservoir Model, which on the surface seems minimalistic, but which displays some unusual behaviour of having a dual system of short-term and long-term memory. Hence we named it **Dual Memory QRC (DM-QRC)**.*



QRFM parameterisation
(ZZ / IQP feature map vs RX angle encoding)

Note: The PennyLane-style IQP feature map is very similar to the Qiskit-style ZZ feature map.

$$U_{\mathrm{IQP}}(x) = \left[\prod_{j<k} \mathrm{CNOT}_{j,k} R_{Z_k}(x_j x_k) \mathrm{CNOT}_{j,k}\right] \cdot \left[\bigotimes_{j=1}^{n} R_{Z_j}(x_j)\right]$$

The distinction comes from their origin. IQP provides advantage to scientific simulations (linear relationship between data magnitude and rotation), while ZZ feature map offers benefit to ML applications (stretches and warps data manifolds to gain expressivity).

$$U_{enc}^{ZZ}(\mathbf{x}(t), \mathbf{b}) = \prod_{k>j} CNOT_{(j,k)} \cdot RZ_k\Big(-2(\pi - x_j(t) - b_j)(\pi - x_k(t) - b_k)\Big) \cdot CNOT_{(j,k)}$$

$$RX(u_j(t) + b_j)$$

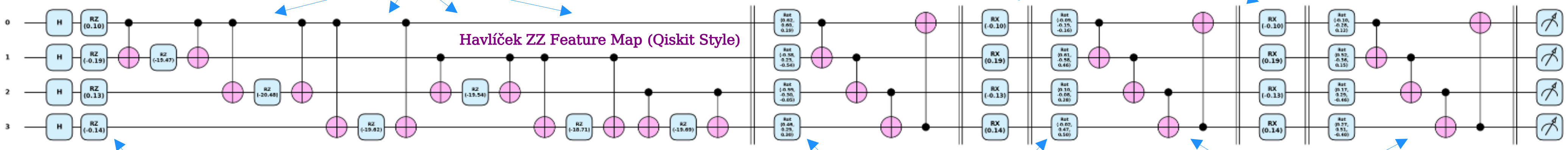
second-order input terms (with biases)

data reuploading (with biases)

$\langle X_1\rangle, \langle Y_1\rangle, \langle Z_1\rangle$

$\langle X_0\rangle, \langle Y_0\rangle, \langle Z_0\rangle$

Havlíček ZZ Feature Map (Qiskit Style)



$\langle X_3\rangle, \langle Y_3\rangle, \langle Z_3\rangle$

$\langle X_2\rangle, \langle Y_2\rangle, \langle Z_2\rangle$

first-order input terms (with biases)

Pauli-Z/Y/Z rotations of weights

entanglement blocks

$$U_{enc}^{Z}(\mathbf{x}(t), \mathbf{b}) = \prod_{j=1}^{N} RZ_j\big(-2x_j(t) - 2b_j\big)$$

$$R_Z(\omega)\, R_Y(\theta)\, R_Z(\phi)$$

$$U_{\mathrm{ent}} = \prod_{i=1}^{N_q-1} \mathrm{CNOT}_{i,i+1}$$

single-qubit Pauli expectations

Note: as model training is sensitive to the magnitude of its parameters, model training is tuned by the number of circuit qubits and layers, as well as the scale of weights, biases and phase shifts (hyper-parameters).

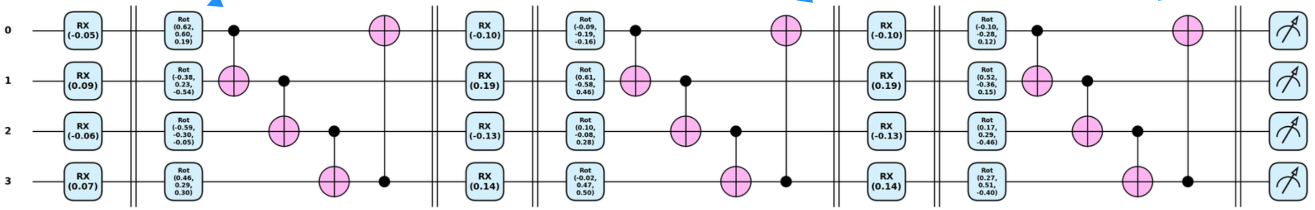
Additional measures have been devised to group two categories of hyper-parameters:

- ***Structural Complexity*** as a proxy for the circuit size ≈ the number of trainable parameters (qubits# * layers#)
- ***Hyper-Scale Magnitude*** as a proxy for the total volume of circuit parameters (weight scale x bias scale * phase scale)

Pauli-Z/Y/Z rotations of weights

entanglement blocks

Angle Encoder (Rx)



$\langle X_0\rangle, \langle Y_0\rangle, \langle Z_0\rangle$

$\langle X_1\rangle, \langle Y_1\rangle, \langle Z_1\rangle$

$\langle X_2\rangle, \langle Y_2\rangle, \langle Z_2\rangle$

$\langle X_3\rangle, \langle Y_3\rangle, \langle Z_3\rangle$

Pauli-X encoded terms (with biases)

data reuploading (with biases)

$$RX(u_j(t) + b_j)$$

$$RX(u_j(t) + b_j)$$

Quantum Reservoir Model

Note: no reuploading shown here, also order of operators right to left

- **Input / Output**
  - Let the original time series be denoted by $S = \{s1, s2, \ldots, sN\}$.
  - The QRC data is prepared using a sliding window of size w.
  - Input $\{u(t)\}$ and output vectors $y(t)$ at time t form a dataset $\mathcal{D}_S = \{(\mathbf{u}(t), y(t))\}$.

$$\mathbf{u}(t) = [s_t, s_{t+1}, \ldots, s_{t+w-1}]^T \in \mathbb{R}^w \qquad y(t) = s_{t+w} \in \mathbb{R}$$

- **Classical Echo State Network (ESN) Layer**
  - or a given input window $\boldsymbol{u}(t)$, a classical state vector $\boldsymbol{x}^{cl}(t)$ is calculated at each time step t. This classical state is updated using a simplified *leaky integrator equation*, where $\gamma \in (0,1)$ is a *leakage rate*:

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

- **Input Encoding Block $U_{enc}(x(t), b)$**
  - Input is encoded into the quantum state via unitary operations, e.g. using *angle embedding*, *ZZ feature map* or a similar *IQP embedding* block.

    In all schemes, random bias terms $\boldsymbol{b}$ are added to the classical state vector $\boldsymbol{x}^{cl}(t)$ before it is encoded into the initial state of the quantum register.

    In angle embedding, all qubits are encoded with rotational gates, e.g. we apply the RX gate to the j-th qubit:

$$U_{\text{enc}}(\mathbf{x}(t), \mathbf{b}) = \bigotimes_{j=1}^{N_q} RX_j(x_j(t) + b_j)$$

While using Havlíček / Qiskit style ZZ feature map for an N-dimensional classical data vector $\boldsymbol{x} = (x_1, x_2, \ldots, x_N)$, the ZZ feature map consist of two components, the first and second order terms of RZ gates (the latter applies only to circuits of two or more qubits):

$$U_{enc}(\mathbf{x}(t), \mathbf{b}) = U_{enc}^{ZZ}(\mathbf{x}(t), \mathbf{b}) \cdot U_{enc}^{Z}(\mathbf{x}(t), \mathbf{b})$$

where:

$$U_{enc}^{Z}(\mathbf{x}(t), \mathbf{b}) = \prod_{j=1}^{N} RZ_j\big(-2x_j(t) - 2b_j\big)$$

$$U_{enc}^{ZZ}(\mathbf{x}(t), \mathbf{b}) = \prod_{k>j} CNOT_{(j,k)} \cdot RZ_k\Big(-2(\pi - x_j(t) - b_j)(\pi - x_k(t) - b_k)\Big) \cdot CNOT_{(j,k)}$$

- **Quantum Reservoir Feature Map**
  - The complete circuit U consists of the encoding and the parameterised reservoir blocks, zero initialised ($|0\rangle^{\otimes n}$).

$$U(\mathbf{x}(t), \mathbf{W}, \mathbf{b}) = U_{res}(\mathbf{x}(t), \mathbf{W}) \cdot U_{enc}(\mathbf{x}(t), \mathbf{b})$$

- **Parameterised Reservoir Block $U_{res}(W)$ (reuploading can be added)**
  - This block of L layers consists of entangling and rotational unitaries, parameterised by a tensor $\boldsymbol{W}$ of random but fixed weights $[\phi, \theta, \omega]$:

$$U_{\text{res}}(\mathbf{W}) = \prod_{l=1}^{L} U_{\text{layer}}^{(l)}(\mathbf{W}) \qquad U_{\text{layer}}^{(l)}(\mathbf{W}) = U_{\text{ent}} U_{\text{rot}}^{(l)}(\mathbf{W}) \qquad U_{\text{ent}} = \prod_{i=1}^{N_q-1} \text{CNOT}_{i,i+1}$$

$$U_{\text{rot}}^{(l)}(\mathbf{W}) = \bigotimes_{i=1}^{N_q} U_{\text{rot}}(\phi_{l,i}, \theta_{l,i}, \omega_{l,i}) \qquad U_{\text{rot}}(\phi, \theta, \omega) = R_Z(\omega)\, R_Y(\theta)\, R_Z(\phi)$$

- **Final Reservoir State q(t)**
  - The reservoir final state $\boldsymbol{q}(t)$ is is a 3N dimensional and is constructed by measuring single-qubit Pauli expectation values $\{\sigma_x, \sigma_y, \sigma_z\}$:

$$\mathbf{q}(t) = \left[\langle\sigma_x^{(1)}\rangle, \ldots, \langle\sigma_x^{(N_q)}\rangle, \langle\sigma_y^{(1)}\rangle, \ldots, \langle\sigma_y^{(N_q)}\rangle, \langle\sigma_z^{(1)}\rangle, \ldots, \langle\sigma_z^{(N_q)}\rangle\right]^T$$

where: $\langle\sigma_p^{(i)}\rangle = \langle\psi(\mathbf{x}^{\text{cl}}(t))|\, \sigma_p^{(i)}\, |\psi(\mathbf{x}^{\text{cl}}(t))\rangle$

- **Readout Layer (the same as in the CRC)**
  - A classical ridge regression maps effective state to prediction $\hat{y}(t)$:

$$\hat{y}(t) = \mathbf{W}_{\text{out}}^T \mathbf{q}(t)$$

where weights w and biases b are trained with regularisation parameter λ:

$$\mathbf{W}_{\text{out}} = (\mathbf{R}^T\mathbf{R} + \lambda\mathbf{I})^{-1}\, \mathbf{R}^T\, \mathbf{Y}_{\text{target}}$$

by minimising the regularised mean square error with the cost function:

$$\mathcal{L}(\mathbf{W}_{\text{out}}) = \|\mathbf{R}\mathbf{W}_{\text{out}} - \mathbf{Y}_{\text{target}}\|_2^2 + \lambda\|\mathbf{W}_{\text{out}}\|_2^2$$

Dual Memory QRC vs.
other CRC / QRC types

So many approaches...

CRC: ESN with leaky integrator (Jaeger 2001/2004), next-generation (Gauthier 2021), time-delay (Appeltant 2012), liquid state machines (Maass 2011), physical (Stepney 2024), extreme learning machines (Huang 2004), ...

QRC: feedback-driven (Kobayashi 2024), dissipative (Sannia 2024), multivariate (Hamhoum 2025), QELM (Oinnocenti 2023), ...

Significant developments in the RC evolution:

- **Jaeger (2001/2004): ESN with Leaky Integrator.** The leaky integrator introduces the leakage rate γ, which acts as a low-pass filter to control the system's "memory depth."
- **Maass (2002): Liquid State Machines (LSM).** LSMs showed that reservoirs don't have to be continuous. They can be a sparse, with fading-memory "liquid" of spiking neurons.
- **Appeltant et al (2011): Time-Delay Reservoir Computing.** TDRC provides a structural turning point. Instead of thousands of physical nodes, it multiplexes a single nonlinear node in the time domain using delay loops. It demonstrates that RC could exploit complex temporal relationships.
- **Gauthier (2021): Next-Generation RC.** NGRC demonstrates that a random, high-dimensional recurrent network of an ESN can be entirely replaced by an explicit, non-recurrent polynomial feature map of time-delay vectors. NGRC shows that the "reservoir" is essentially just a clever nonlinear feature mapping step.
- **Also, … Stepney (2024): Physical RC Models.** Because the reservoir weights are never trained, any sufficiently complex physical system can act as a reservoir, e.g., optical cavities, spintronic oscillators, or bucket of water waves.

Ideas leading towards the Dual Memory QRC

ESN is key! Yet, it is responsible for long/short term memory. Also, input one step at a time. Leakage controls only internal state decay. *Flow:*

Input(t) $\rightarrow$ Leaky Temporal Blending $\rightarrow$ Recurrent State Update h(t) $\rightarrow$ Readout.

Sparse reservoirs are sufficient.

Seek new ways of dealing with temporal relationships. *Flow:*

Input(t) $\rightarrow$ Temporal Masking (Sample&Hold) $\rightarrow$ Nonlinear Node with Delay Loop $\rightarrow$ Readout.

Explicit nonlinear feature mapping can replace random networks. Uses TS windows! However, no internal reservoir / no long-term memory balance. Also, poly combinations explode combinatorially! *Flow:*

Input(t) $\rightarrow$ Time-Delay Embedding Matrix $\rightarrow$ Polynomial Transformations $\rightarrow$ Readout.

Dual Memory QRC (2025):

Adopts Leaky ESN as its temporal memory. Quantum Feature Map is used for high-dimensional mapping and temporal spacialisation. Quantum phase warps the reservoir manifold, isolating relations. *Flow:*

Input(t) $\rightarrow$ Leaky Combination $\rightarrow$ Quantum Feature Map $\rightarrow$ Measurement / Readout.

Why quantum reservoir?
NN vs. QNN

To understand where Quantum Reservoir Computing is heading, we have to look at how a classical Deep Neural Network and a Quantum Neural Network shape data geometrically.

Deep classical networks are spectacularly successful because they build layers of hierarchical feature abstraction.

By stacking linear combinations with non-linear activations, they iteratively warp the Euclidean manifold, pulling apart complex correlations in the data layer by layer.

This compositional approach is highly effective, but it requires optimizing thousands or millions of parameters across a deep architecture to construct those feature boundaries.

Feature	**Classical Neural Network (NN)**	**Quantum Neural Network (QNN)**
Fundamental Manifold	Euclidean $\mathbb{R}^n$ (warped by activations)	Complex Projective Space $\mathbb{C}P^{2^n-1}$ (warped by equiv. rel.)
Normalization Strategy	Local/Layer (e.g., Softmax, BatchNorm)	Global (Unitary L_2 norm preservation)
Equivalence Relation	Scale Invariance in ReLU nets $(ReLU(a \cdot x) = a \cdot ReLU(x) \mid a > 0)$	Global Phase Invariance $(\lvert\psi\rangle \sim e^{i\phi}\lvert\psi\rangle)$
Metric (Parameter)	Euclidean L_2 Distance	Fubini-Study Metric
Information Geometry	Fisher Information Matrix (FIM)	Quantum Fisher Information Matrix (QFIM)
Loss Landscape	Non-convex, affine-based transformations	Periodic, defined by $SU(2^n)$ rotations
State Representation	Bounded hypercube or open vector space	Generalization of the n-qubit Bloch Sphere

A Quantum Reservoir offers an entirely different geometric mechanism.

We use a quantum feature map to project our memory-blended classical state directly into the exponentially vast Hilbert space.

The trajectories are warped through unitary transformations, quantum phase relations, and entanglement.

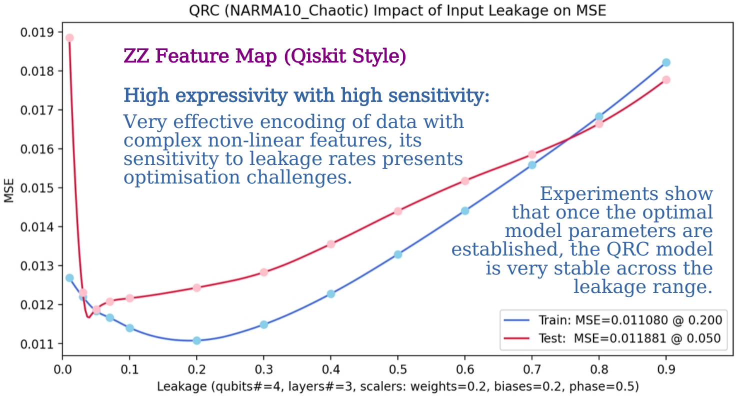
This creates a phase-warped manifold where relationships between time-steps are mapped to quantum state amplitudes.

QNNs leverage the natural, massive dimensionality of Hilbert space to untangle complex, chaotic dynamics.

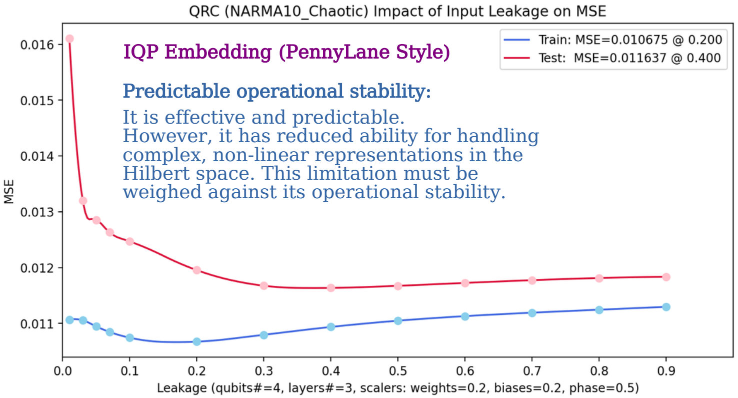
QRC data encoding

ZZ, IQP or Angle encoding?

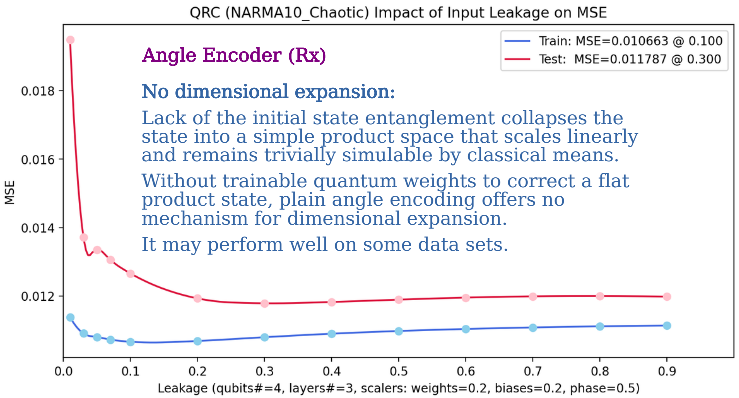
QRC models are highly sensitive to the methods of their data encoding and scaling of their weights, biases and inputs.



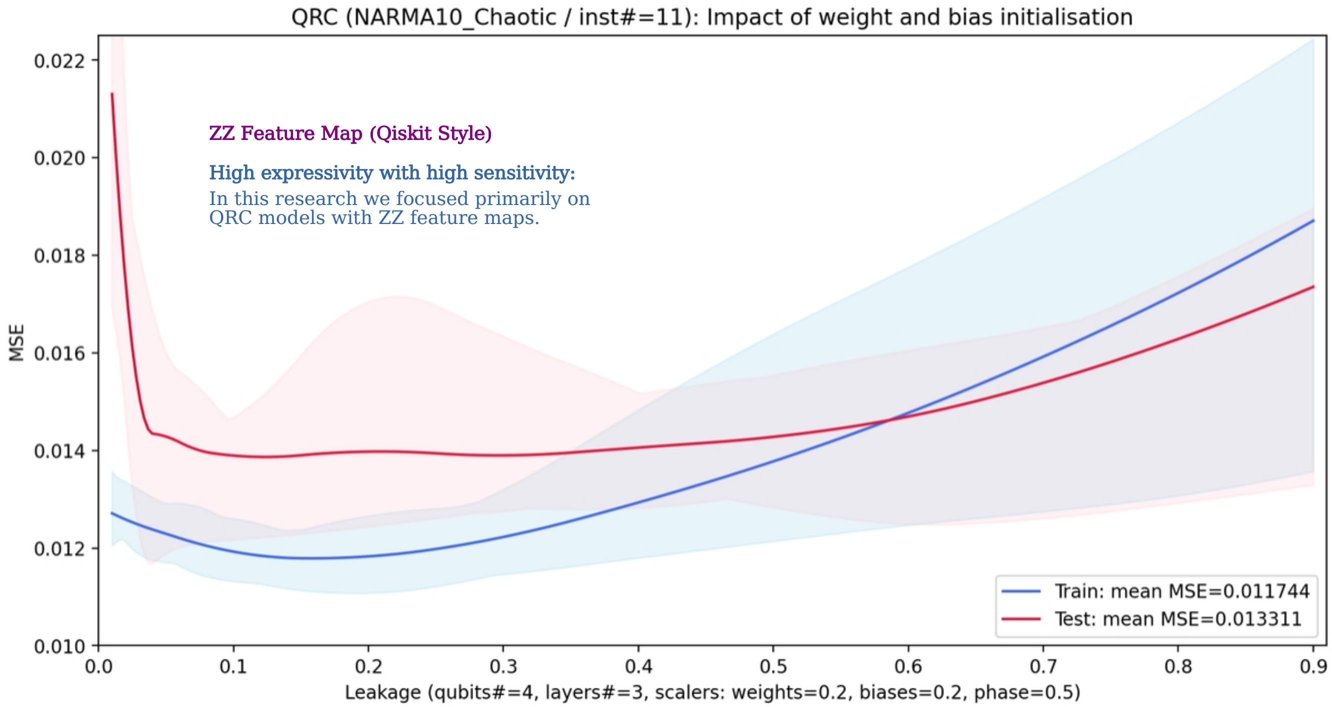
- Embedding time-series values directly into the quantum phase potentially creates a data-warped manifold in the Hilbert space.
- This allows projecting input data into a high-dimensional correlated state space, without any further training of the QRC weights.
- This ensures that correlations between the sliding input window and the reservoir states can also be established.
- The exact nature of this geometric warping depends on the embedding topology of a quantum data encoder.
  - **Qiskit-style, Havlíček ZZ feature maps** introduce data-dependent entangling phases that capture localised, pairwise feature interactions, which produces strong manifold warping.
  - **PennyLane-style IQP (Instantaneous Quantum Polynomial-time) embedding** utilises multi-layered, cascaded phase shifts to construct deep, smooth multi-body interference patterns.



- This distinction dictates how the leakage rate (α) regulates temporal memory in volatile time series.
- Such a control is vital to the reservoir's resilience to phase scrambling.
- **Angle encoding** merely offers linear transformation of input only – little value against classical methods.



QRC Sensitivity
to initialisation of weights and biases



- QRC models are sensitive to their architectural complexity.
- Therefore, in this study we evaluated model architectures by varying their numbers of qubits, layers, leakage ratios, and scaling constants (for weights, biases and phases).

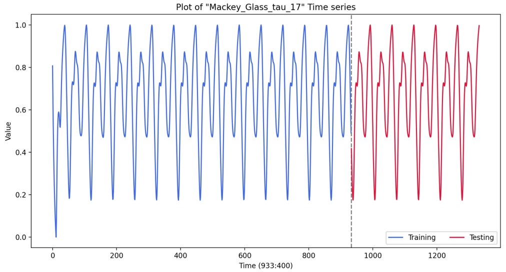
  For each data set, we tested 3240 QRC and 720 CRC model architectures.
- QRC models also to sensitive to the initialisation of their weights and biases.
- Therefore, in this study performance of each model was evaluated as the mean performance of 11 differently initialised model instances.

  Therefore, for each data set we developed and tested a total of 261,360 model instances.
- We then applied each model instance to 6 different chaotic data sets.

Time series in use
Chaotic data generators

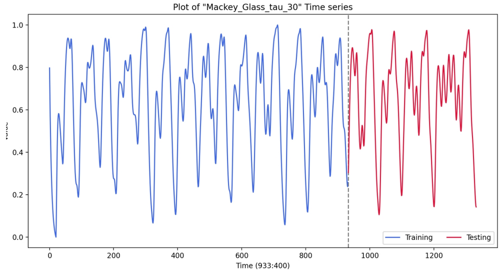
Three chaotic data generators were in use, i.e.

- **Mackey-Glass:** *Continuous-Time Delayed Chaos* (Deterministic & Autonomous)
- **NARMA:** *Discrete-Time Non-Linear Stochastic Process* (Random Noise + Non-linear Filter)
- **ARMA:** *Linear Stochastic Process / Filtered White Noise* (Random Noise + Linear Filter)



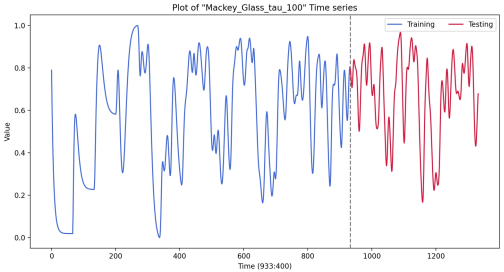
Mackey-Glass 17

```
Lag used:                          7
Largest Lyapunov Exponent (λ₁):  0.0018
Correlation Dimension (D₂):      1.1200
Permutation Entropy (H):         0.9197
Interpretation:
  Indeterminate / Borderline Chaos
```



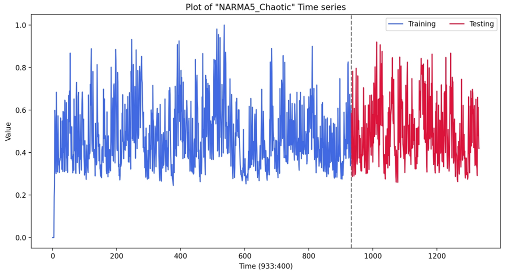
Mackey-Glass 30

```
Lag used:                         10
Largest Lyapunov Exponent (λ₁):  0.0176
Correlation Dimension (D₂):      1.6236
Permutation Entropy (H):         0.9556
Interpretation:
  High-Dimensional Chaos / Weakly Constrained Map
```



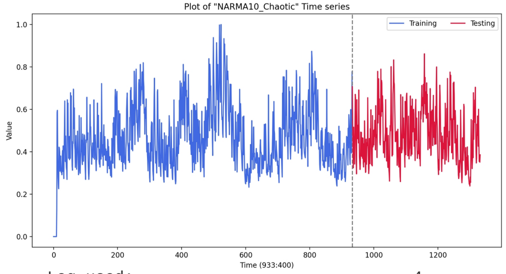
Mackey-Glass 100

```
Lag used:                         20
Largest Lyapunov Exponent (λ₁): -0.0377
Correlation Dimension (D₂):      1.6427
Permutation Entropy (H):         0.9961
Interpretation:
  Linear Stochastic / Noise
  (High Entropy, Flat Trajectory)
```



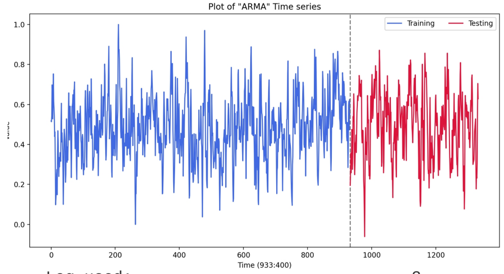
NARMA 5

```
Lag used:                          2
Largest Lyapunov Exponent (λ₁): -0.0002
Correlation Dimension (D₂):      1.1098
Permutation Entropy (H):         0.9964
Interpretation:
  Linear Stochastic / Noise
  (High Entropy, Flat Trajectory)
```



NARMA 10

```
Lag used:                          4
Largest Lyapunov Exponent (λ₁):  0.0134
Correlation Dimension (D₂):      1.4979
Permutation Entropy (H):         0.9991
Interpretation:
  High-Dimensional Chaos / Weakly Constrained Map
```



ARMA

```
Lag used:                          3
Largest Lyapunov Exponent (λ₁):  0.0042
Correlation Dimension (D₂):      2.0985
Permutation Entropy (H):         0.9976
Interpretation:
  High-Dimensional Chaos / Weakly Constrained Map
```

The generated time series were characterised using the following metrics:

- **Lyapunov Exponent:** *Rate of Divergence* Measures the exponential rate at which two infinitesimally close trajectories in phase space separate over time, where a positive value confirms deterministic chaos.
- **Correlation Dimension:** *Fractal Spatial Complexity* Quantifies the geometric dimensionality and fractal scaling of a system's chaotic attractor by measuring how the number of data points within a given distance changes as that distance shrinks.
- **Permutation Entropy:** *Ordinal Dynamics Information* Calculates the information extraction rate and unpredictability of a time series by evaluating the frequency and variety of relative order patterns (permutations) among neighbouring data points.

Participation Ratio

What is it?

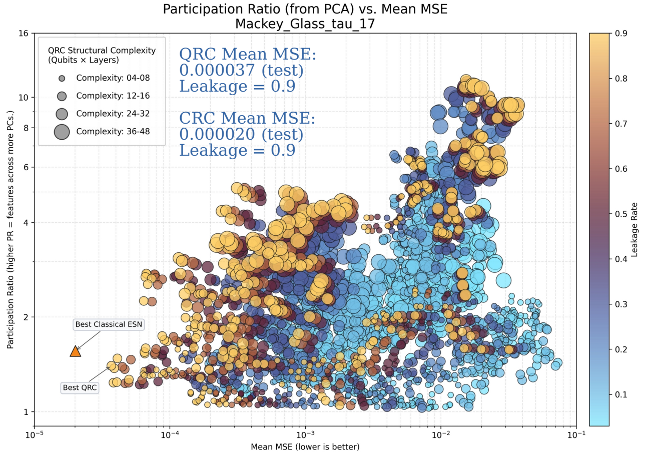
Let's explain the participation ratio first...

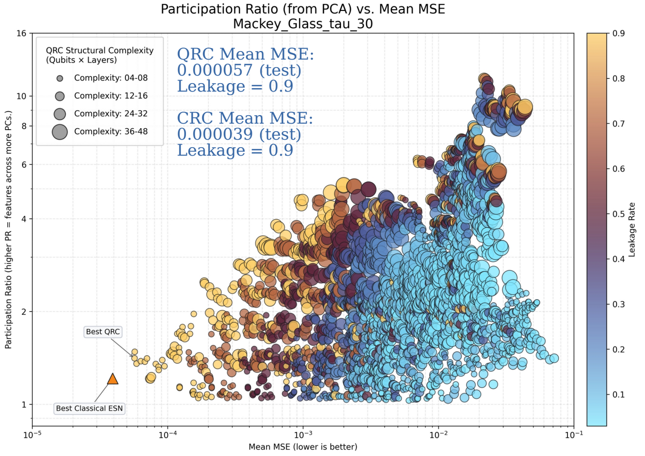
Participation Ratio (PR): *Effective Dimensionality Index*

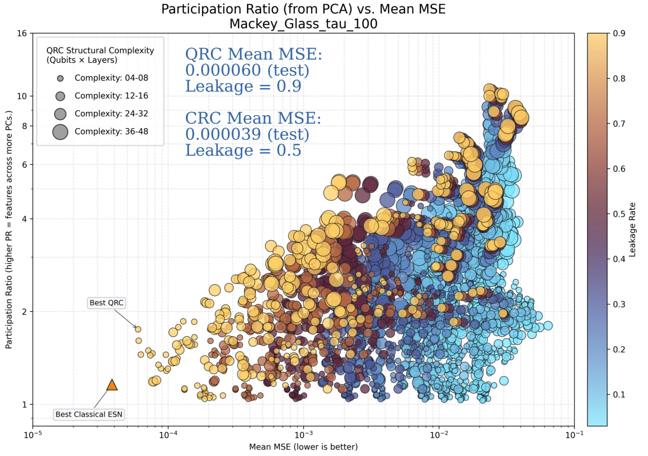
- Measures the effective geometric dimensionality of a dataset by calculating the uniformity of its Principal Component Analysis (PCA) eigenvalue distribution; a high/low value indicates the data variance is spread across many / few principal components.
- In QRC we measure PR between the QRC feature map and the classical readout.

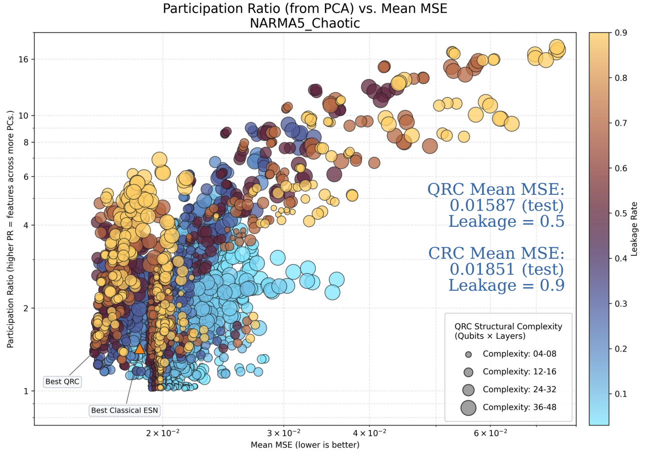
$$\mathrm{PR} = \frac{\left(\sum \lambda_i\right)^2}{\sum \lambda_i^2}$$

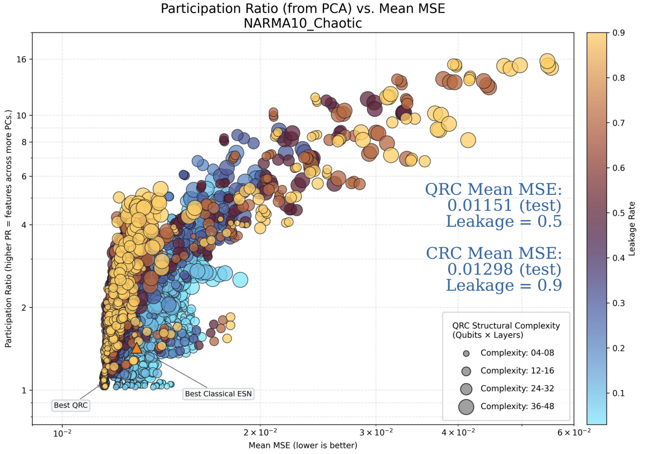
Where λ_i represents the eigenvalues of the data covariance matrix.

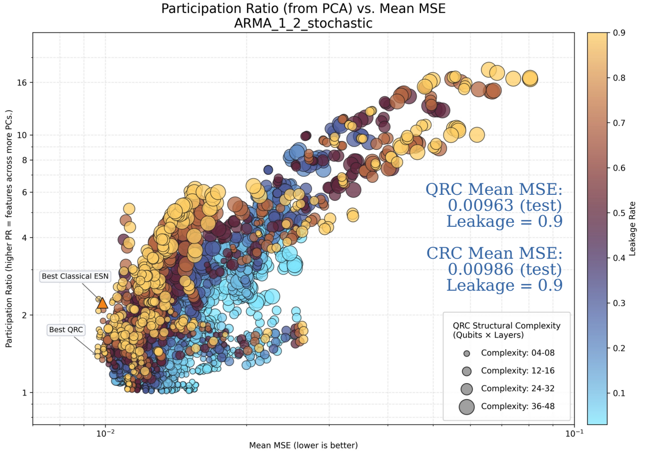






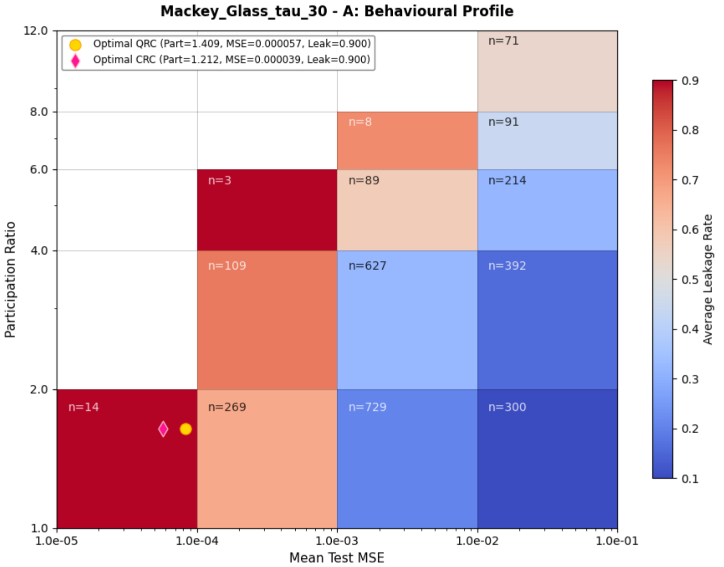


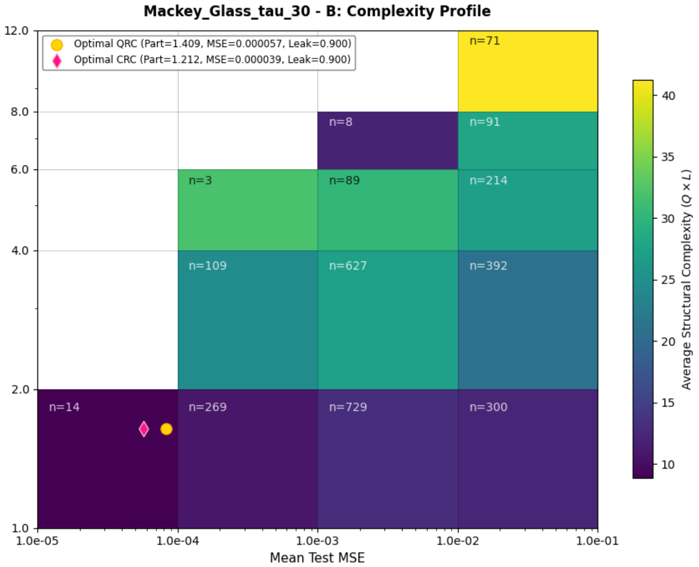


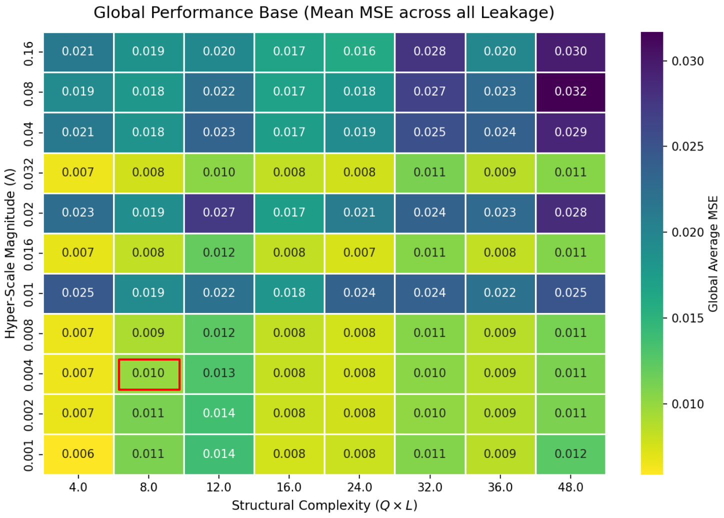


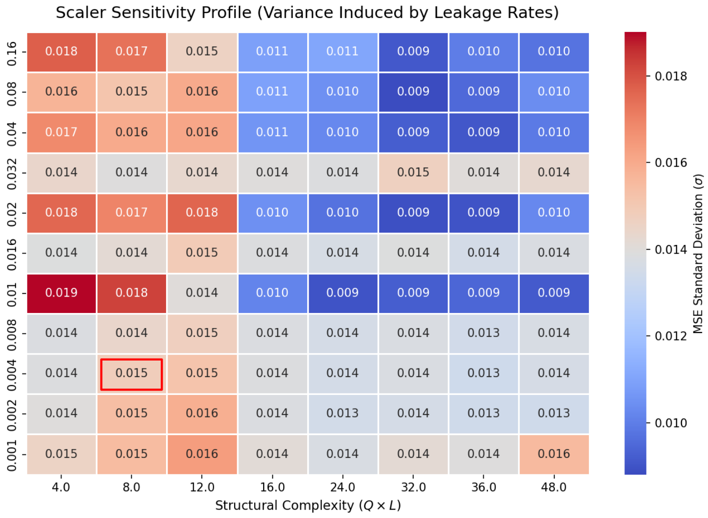
Mackey-Glass (tau=30) model

Let's explain the analytic method first...









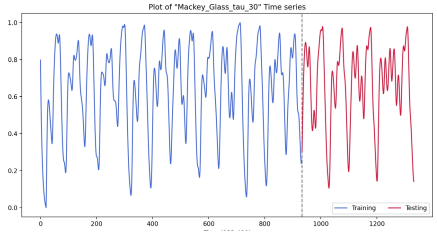
Analysis

Mackey-Glass (tau=30) generates series which are aperiodic, with fluctuations from time-delayed, nonlinear negative feedback.

At tau=30, its quasi-periodic structure is lost (as in tau=17). In spite of its irregularity, the signal can still be understood within a short window. Thus, neither the optimum CRC or QRC are using the reservoir.

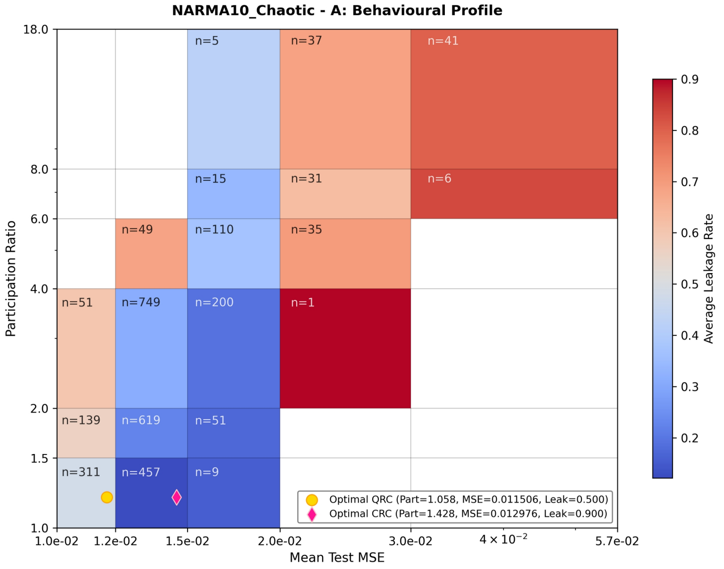
CRC and QRC optimum results are close. QRC best MSE are in the area of small structural complexity and low scaling magnitude.

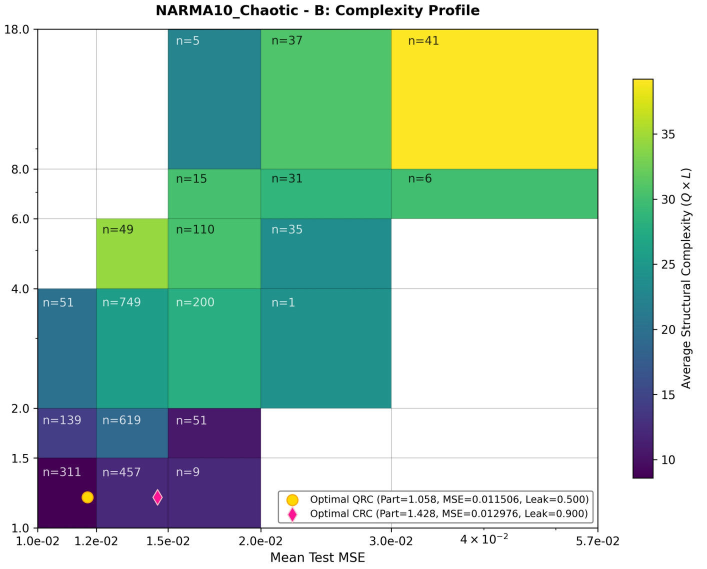
However, the QRCs show low stability of their best results (medium-high MSE variance).



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

NARMA (order=10) model





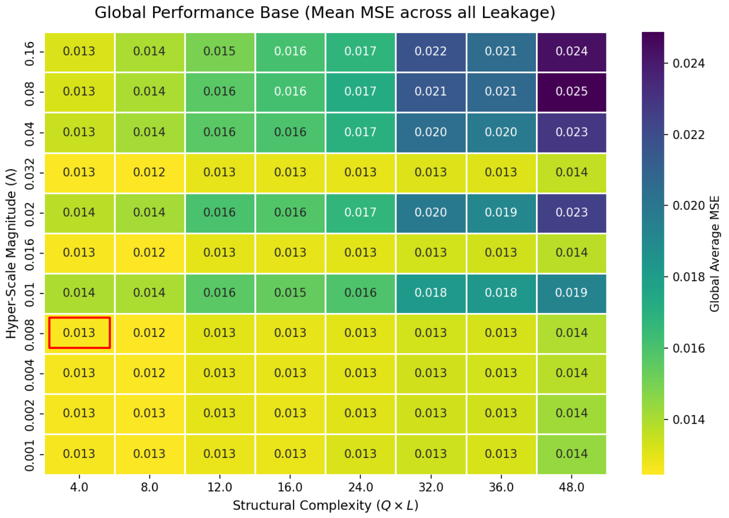
Analysis

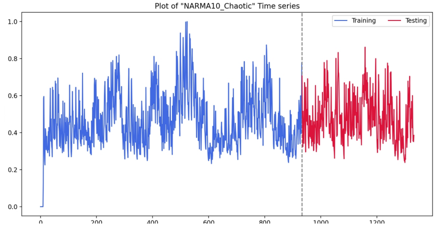
NARMA10 is a series of long-term patterns (10 steps), which are disrupted by chaotic oscillations.

Due to high volatility in a small training dataset, CRC cannot identify patterns in its reservoir. It abandons it, in favour of windows only.

QRC, however, combines the reservoir memory with the window and maps it into high-dim, warped Hilbert space manifold to recognise patterns longer than window's length!

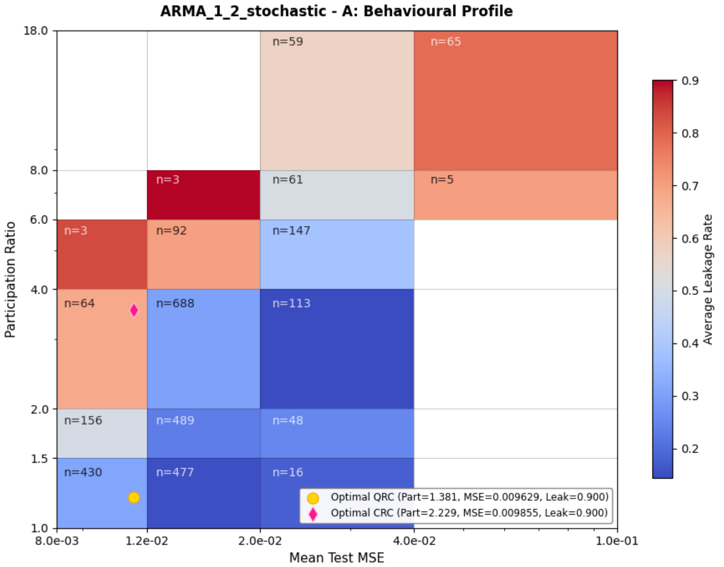
QRC produces its superior results in the area of small complexity and lower hyper-scale magnitude, with stable MSE across all leakage rates.

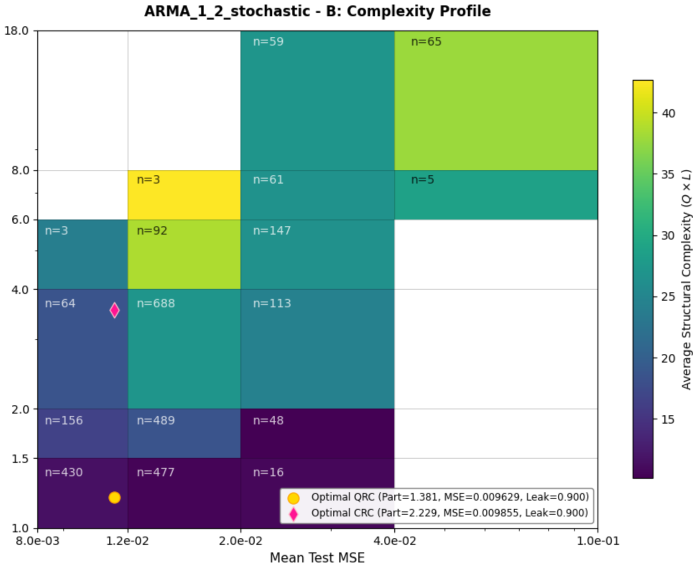


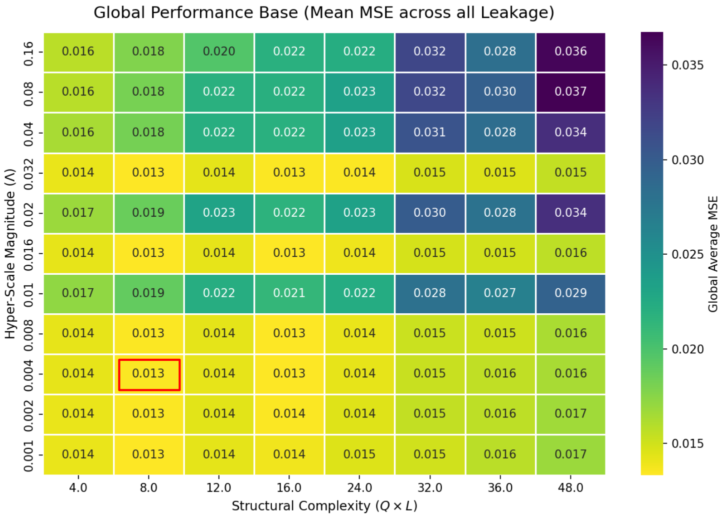


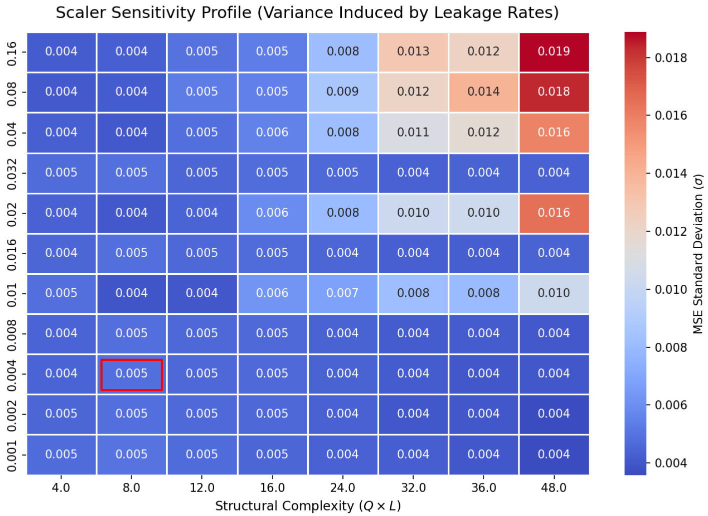
$$\mathbf{x}^{cl}(t) = (1-\gamma)\mathbf{x}^{cl}(t-1) + \gamma \mathbf{u}(t)$$

ARMA model









Analysis

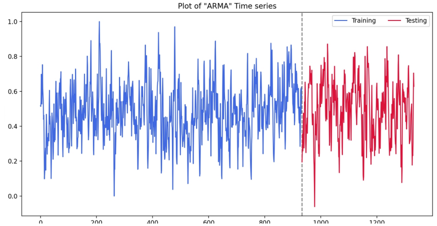
ARMA shows a stationary stochastic process, resulting from a linear combination of past and present values (autoregressive) with white noise (moving average).

Both CRC and QRC relied on their windows and had no need to refer to the reservoir.

QRC best MSE was slightly better than CRC, achieved with smaller models and higher series compression.

In QRC high-performance area, its models were stable across all leakage ratios.

Based on the leakage ratio, the QRC is an outlier in its own participation class.



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

Reflections on experimental results

- **Information embedding in phase-encoded systems.** In a phase-encoded quantum system, information isn't just stored as an amplitude - it is embedded in the high-dimensional interference patterns of the quantum state.
- **Phase information is notoriously sensitive.** In a volatile input sequence, such as those used in experiments, each new time step applies a completely different, heavily warped phase shift to the system.
  - If the leakage rate is low, the quantum state tries to hold onto the previous states' phases.
  - As inputs are changing rapidly and non-linearly, the successive phase shifts quickly cross-modulate, leading to destructive interference or rapid phase scrambling.
  - The solution adopted by QRC models is to raise the leakage high for the system to flush out the historical phase at each step, relying strictly on its immediate non-linear mapping capability from the current window.
- **Optimal QRCs are rarely linked to higher dimensions** - as reflected in their participation ratio.
- **Optimum QRCs are often outliers** in their class of models with respect of their participation ratio, hyper-scale magnitude, structural complexity, and of course performance. Hyper-parameter tuning is thus required!

- **The selection of Havlíček ZZ feature map is vital in a unoptimized reservoir settings.**

  Because we cannot train the quantum layers, we must therefore rely on a feature map that inherently forces maximum non-linear expressivity and multi-body entanglement in the model's forward pass.
- The (pi - x_i)(pi - x_j) coordinate transformation acts as a hard-coded geometric optimiser.

  It spreads the unoptimised classical reservoir states across the Hilbert space automatically, creating the highly warped, linearly separable manifold that allows your classical readout to achieve high quality results.
- **Angle Encoding:**

  Zero entanglement → Linear scaling → No Quantum Advantage
- **IQP Encoding:**

  Smooth entanglement → Exponential scaling → Predictable and tunable

  However, it lacks the non-linear sharpness for highly chaotic datasets.
- **Havlíček ZZ Encoding:**

  Inverted, multiplicative entanglement → Exponential scaling → Massive Quantum Advantage

  It is capable of flattening chaotic series, but is highly volatile and requires meticulous leakage, input (phase), weights and biases scaling to control.

Summary and reflections...

About reservoir computing

- Classical RCs (CRC) are derived from recurrent neural networks.
- Reservoir computing utilises a large sparse neural network of random but fixed weights, which transform input into a high-dimensional space, where data can be easily separated (classified) by a simple linear model.
- The reservoir should be able to echo or retain information about past inputs for a short period. However, the reservoir state should fade away over time – memory leakage.
- Classical RCs were successfully deployed in prediction and forecasting, classification of temporal patterns, noise reduction and anomaly detection in signals, and modelling of chaotic systems.

About DM-QRC

- This project explores the potential of QML to improve CRC algorithms.
- Future QRC applications: machine condition monitoring, astronomical observations, marketing, earthquake prediction, EEG or ECG analysis, etc.
- There are many types of classical and quantum reservoir models, e.g. echo-state networks with leaky integrator, feedback-driven, liquid state machine, next generation, time-delay, dissipative, physical CRCs/QRCs.
- This project focuses on a QRC architecture, which demonstrates a dual memory system for short- and long-term retention, hence named: Dual Memory QRC (DM-QRC).

More observations and reflections

- On input, DM-QRC allows entering time series sliding windows.
- It employs a classical reservoir with the leakage update rule used to blend input windows into the reservoir states.
- Subsequently, a quantum reservoir feature map transforms these classical temporal states into spacialised and interrelated quantum states in high dimensional, warped manifold of the Hilbert space.
- Measuring such quantum states results in a high-dimensional representation of input that can be used for training and application of a simple, yet highly effective classical ML model.
- As DM-QRC weights and biases are randomly selected and fixed, the models are able to efficiently process large ML datasets, even when execution takes place on a quantum simulator.

About experiments with DM-QRC

- It was determined that to gain value from DM-QRC, it is needed to encode data in the quantum phase, e.g. by using ZZ feature map.
- Experiments with DM-QRC forecasting indicate that quantum reservoir models are effective when compared against their classical counterparts.
- Their ability to mix long and short term memories is particularly useful in dealing with long patterns, which would normally elude detection of classical systems.
- It was also discovered that quantum reservoir models were smaller and more stable than the classical reservoirs.

Thank you! Any questions?

Acknowledgments

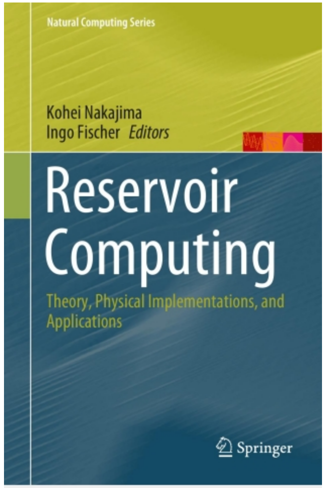
We gratefully acknowledge the support of Paweł Gora, Chief Executive Officer of the Fundacja Quantum AI, whose efforts in organizing and overseeing the AIntern 2025 Program were instrumental in enabling this research.

Selected readings on CRC

- Abbas, A.H., Abdel-Ghani, H., Maksymov, I.S., 2024. *Classical and Quantum Physical Reservoir Computing for Onboard Artificial Intelligence Systems: A Perspective*. Dynamics 4, 643–670. https://doi.org/10.3390/dynamics4030033.
- Gauthier, D.J., Bollt, E., Griffith, A., Barbosa, W.A.S., 2021. *Next generation reservoir computing*. Nat Commun 12, 5564. https://doi.org/10.1038/s41467-021-25801-2.
- Stepney, S., 2024. *Physical reservoir computing: a tutorial*. Nat Comput 23, 665–685. https://doi.org/10.1007/s11047-024-09997-y.
- Viehweg, J., Walther, D., Mäder, P., 2025. Temporal convolution derived multi-layered reservoir computing. Neurocomputing 617, 128938. https://doi.org/10.1016/j.neucom.2024.128938.

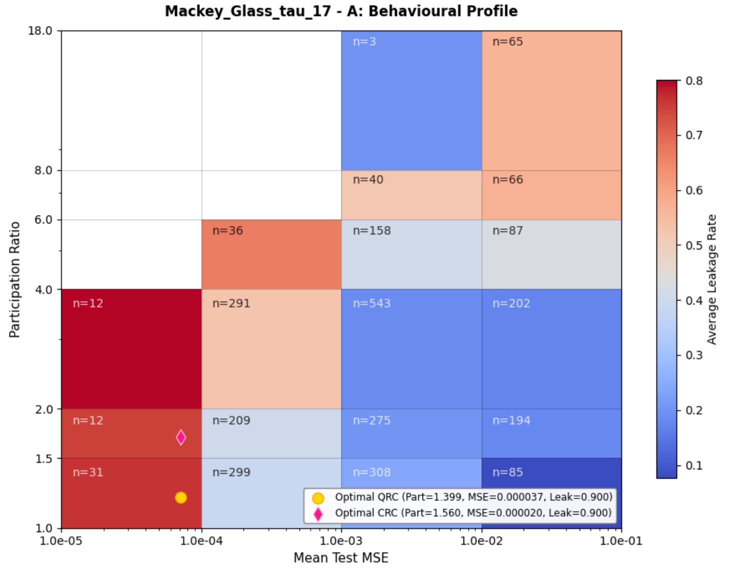
Selected readings on QRC and related QML

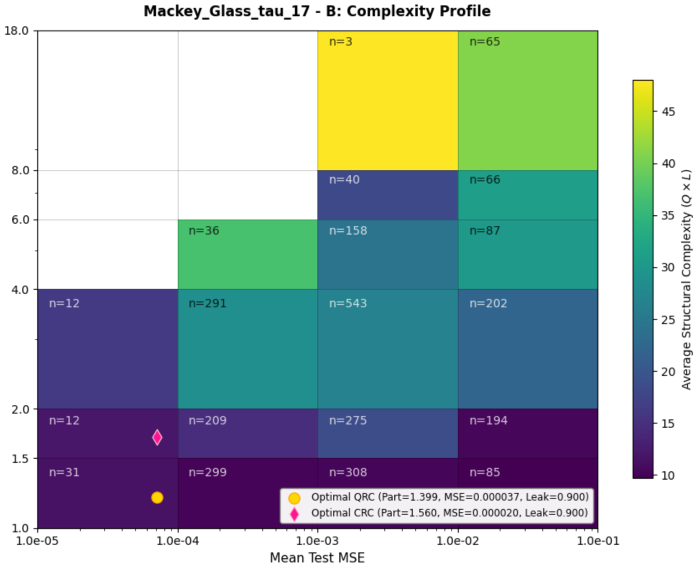
- Chen, J., Nurdin, H.I., Yamamoto, N., 2020. Temporal Information Processing on Noisy Quantum Computers. Phys. Rev. Applied 14, 024065. https://doi.org/10.1103/PhysRevApplied.14.024065.
- Kobayashi, K., Fujii, K., Yamamoto, N., 2024. *Feedback-Driven Quantum Reservoir Computing for Time-Series Analysis*. PRX Quantum 5, 040325. https://doi.org/10.1103/PRXQuantum.5.040325.
- Fujii, K., Nakajima, K., 2020. *Quantum reservoir computing: a reservoir approach toward quantum machine learning on near-term quantum devices*. https://doi.org/10.48550/arXiv.2011.04890.
- Götting, N., Lohof, F., Gies, C., 2023. Exploring quantumness in quantum reservoir computing. Phys. Rev. A 108, 052427. https://doi.org/10.1103/PhysRevA.108.052427.
- 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:2602.21350 (Version 2).

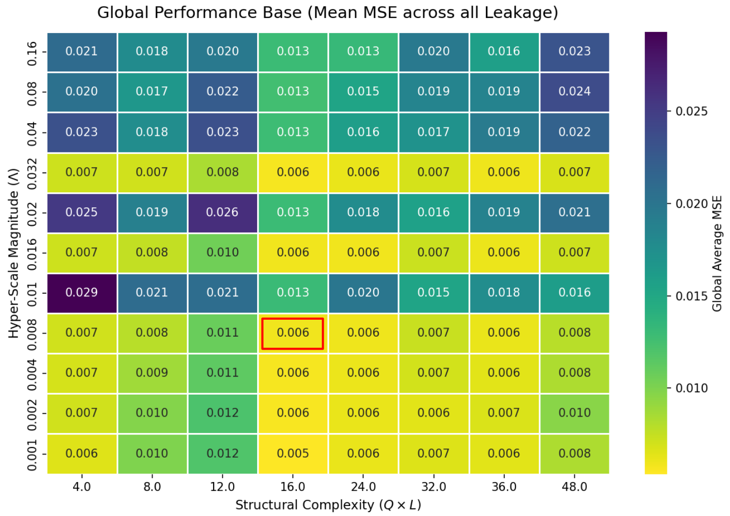


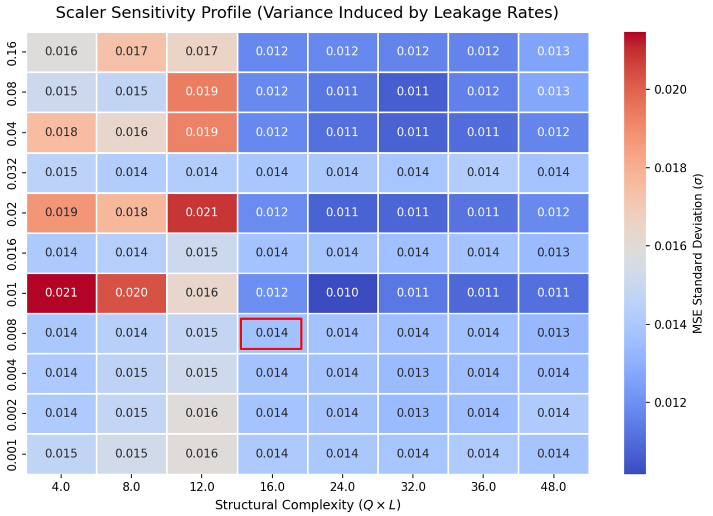
Appendix

Mackey-Glass (tau=17) model









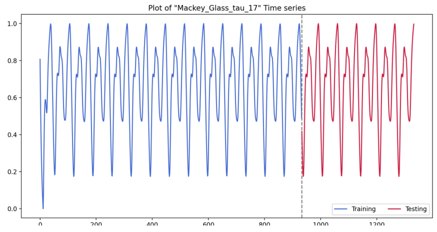
Analysis

Mackey-Glass (tau=17) is a chaotic series, which at tau=17 exhibits a strong underlying pseudo-periodicity. The phase space portrait remains clean, resembling a warped, low-dimensional limit cycle.

Both CRC and QRC achieve great MSE performance even with small models. At high oscillations fitting into short windows, the models did not need to use their reservoirs.

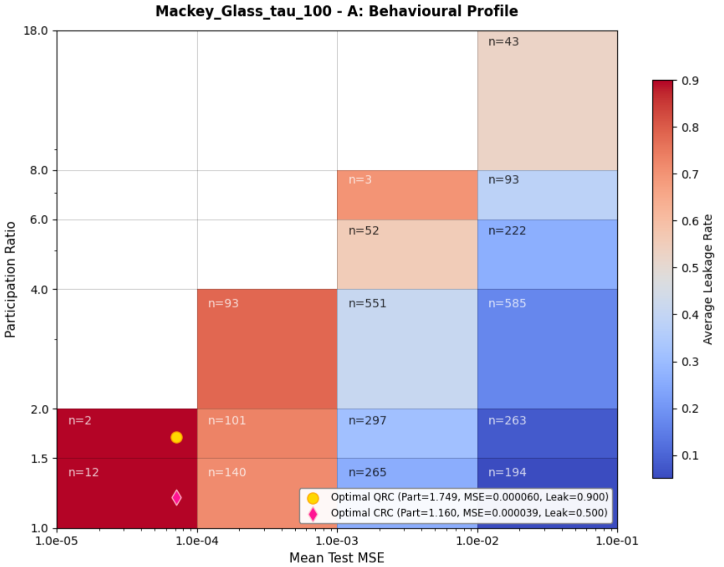
Within high-performance models, QRC was stable across all leakage rates.

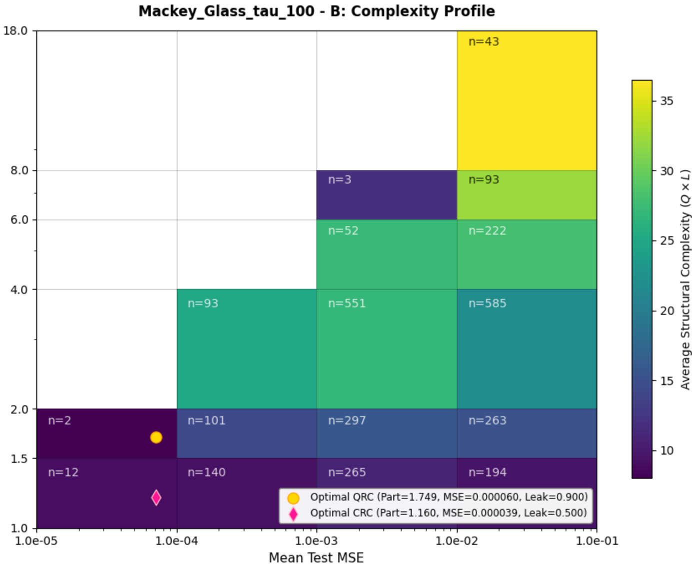
QRC is part of its leakage / participation class.

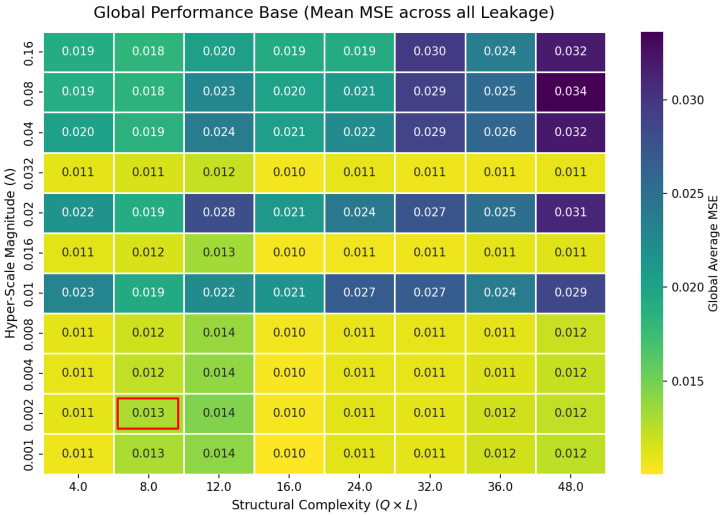


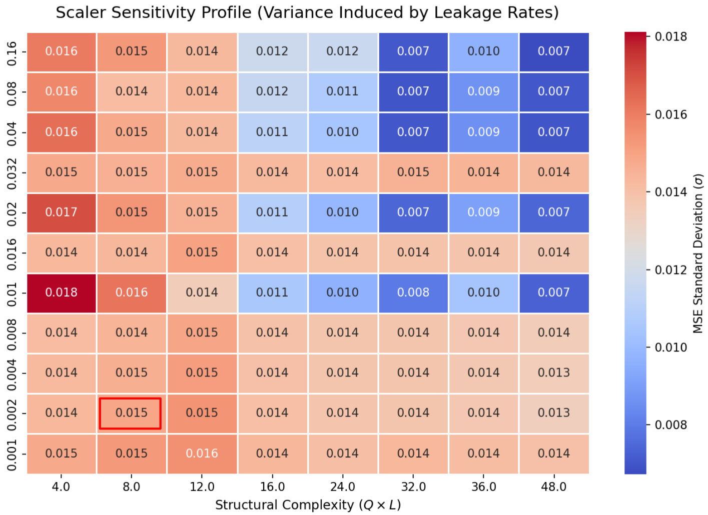
$$\mathbf{x}^{cl}(t) = (1-\gamma)\mathbf{x}^{cl}(t-1) + \gamma\mathbf{u}(t)$$

Mackey-Glass (tau=100) model









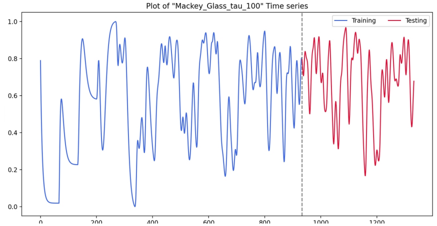
Analysis

Mackey-Glass (tau=100) is aperiodic, with fluctuations from a time-delayed, nonlinear feedback. Its signal shows high turbulence with high permutation entropy.

Both CRC and QRC achieve high MSE performance. Unlike CRC the optimum QRC model did not rely on its reservoir, which explains high volatility of its performance within its class of similarly sized QRC models.

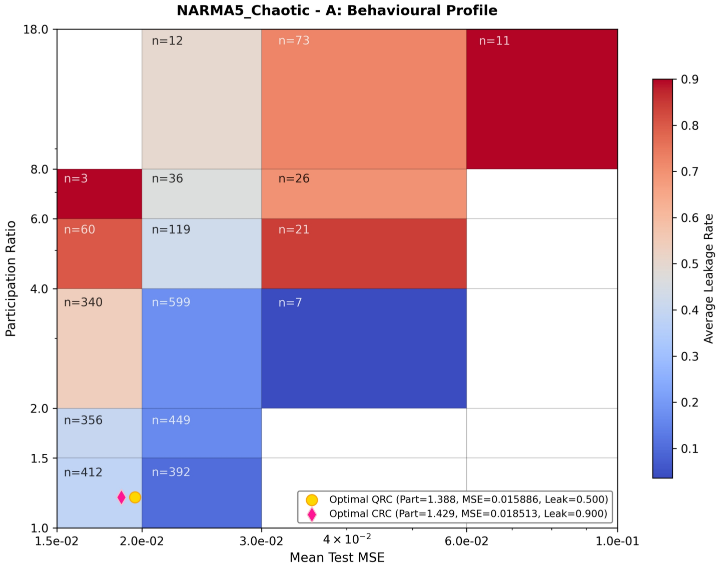
CRC participation ratio is spread across many principal components. Yet, QRC compreses its data efficiently.

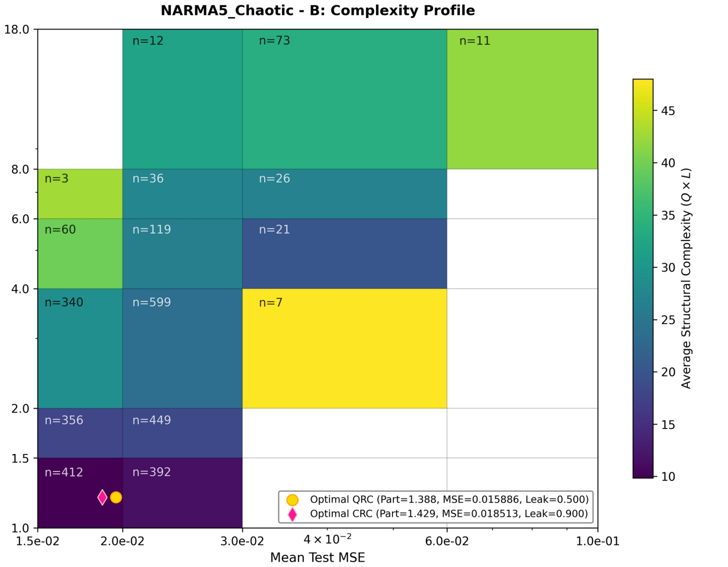
QRC is part of its leakage / participation class.

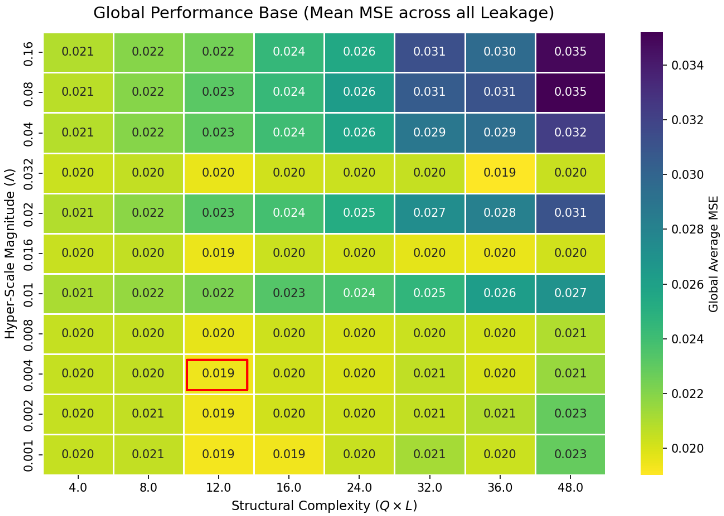


$$\mathbf{x}^{cl}(t) = (1-\gamma)\mathbf{x}^{cl}(t-1) + \gamma\mathbf{u}(t)$$

NARMA (order=5) model







Analysis

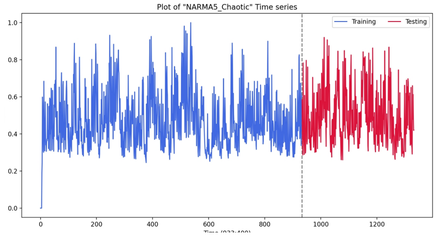
NARMA5 is a series of short-term patterns (5 steps), which are disrupted by oscillations.

Due to the stability of short-term sequences, the CRC performs well, however, uses only its windows data.

QRC on the other hand adopted a medium leakage, relying on both the reservoir and its windows.

QRC optimum results are slightly better than those of the CRC. QRC is in the area of small complexity and low scaling, which are also stable across all leakage rates.

QRC is part of its leakage / participation class.



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