npj Quantum Information

https://doi.org/10.1038/s41534-026-01229-8

Article in Press

Quantum reservoir computing induced by controllable damping

Received: 10 July 2025

Accepted: 22 March 2026

Cite this article as: Ricci, E., Monzani, F., Nigro, L. *et al.* Quantum reservoir computing induced by controllable damping. *npj Quantum Inf* (2026). https://doi.org/10.1038/s41534-026-01229-8

Emanuele Ricci, Francesco Monzani, Luca Nigro & Enrico Prati

Quantum reservoir computing induced by controllable damping

Emanuele Ricci [*], Francesco Monzani [*], Luca Nigro, and Enrico Prati [†]
Dipartimento di Fisica "Aldo Pontremoli", Università degli Studi di Milano, Via Celoria 16, 20133, Milan, Italy

ABSTRACT

Quantum reservoir computing has emerged as a promising machine learning paradigm for processing temporal data on near-term quantum devices, as it exploits the large computational capacity of qubits without suffering from typical issues arising when training variational quantum circuits. In particular, quantum gate-based echo state networks have proven effective when the evolution of the reservoir circuit is non-unital. Nonetheless, a method for ensuring a tunable and stable non-unital circuit evolution was lacking. We propose an algorithm that induces damping by applying a controlled rotation to each qubit in the reservoir. It enables tunable, circuit-level amplitude amplification of the zero state, maintaining the system away from the maximally mixed state and preventing information loss caused by repeated mid-circuit measurements. The algorithm is inherently stable over time, as it can process arbitrarily long input sequences, beyond the coherence time of individual qubits, by inducing arbitrary damping on each qubit. Moreover, we show that quantum correlations between qubits improve memory retention, underscoring the potential utility of a quantum system as a computational reservoir. We demonstrate, through standard reservoir computing benchmarks, that this algorithm enables robust and scalable quantum random computing on fault-tolerant quantum hardware.

INTRODUCTION

Motivated by promising insights into quantum algorithms for machine learning, significant attention has been devoted to their implementation on current quantum computers. However, implementing these algorithms on currently available quantum hardware—still limited by the inherent noise of noisy intermediate-scale quantum (NISQ) devices—remains a major challenge [1–3]. In particular, non-variational algorithms, such as quantum kernel methods [4], quantum principal component analysis [5], quantum feature maps [6], and extreme learning machine [7], which do not require training of internal parameters of the quantum circuit, emerge as successful techniques since they inherently avoid the typical training problems [8, 9]. Among them, quantum reservoir computing stands out as particularly effective in handling sequential data processing for time series analysis, forecasting of chaotic dynamics, anomaly detection, and classification tasks [10].In this work, we present and validate a gate-based quantum reservoir computing algorithm that leverages dissipation induced by an ancillary bath to preserve and regulate information within the internal reservoir dynamics.

Reservoir computing is a well-established supervised machine learning algorithm that employs a fixed neural network to process sequential data, originally stemming from the echo state network [11] and the liquid state machine [12, 13] models. It has emerged as a paramount example of unconventional computing deployment [14, 15], with diverse physical implementations [16]. Among them, many quantum systems – associated with their exponentially large computational capacity – has been harnessed for reservoir computing [17–33].

Focusing on superconducting qubits embodiment of quantum reservoir computing, besides early implementations [34–36], they have recently been empowered by the possibility of mid-circuit measurements [37–39]. Nonetheless, the repeated measurements of the qubit register progressively steer the quantum system toward the maximally mixed state, thereby reducing the expressivity of the network and hindering its ability to distinguish different input values [40]. Thus, the loss of information should be mitigated by inducing a partial purification of the state of the quantum network. In previous works [41–43], amplitude damping has been recognized as an effective computational resource for sequential tasks of quantum reservoir computing. Specifically, these works show how a source of non-unital noise, as already theoretically suggested in Refs. [44, 45] breaks the symmetry of the quantum evolution by biasing the state of the system away from the maximally mixed configuration.

Previously, some of the authors have addressed learning algorithms by exploiting noise for both Boltzmann machines [46] and quantum reservoir computing [35, 41]. Here, we propose an algorithm that induces damping, ensuring non-unital dynamics, without relying on native hardware dissipation. In contrast with previous embodiments on intrinsically noisy quantum hardware [37, 41], such an algorithm, essentially based on controlled rotations of the accessible qubits, allows for a completely tunable damping rate, thus ensuring long-term computations. In particular, our method offers a tunable framework to explore the interplay between memory retention and non-unitality in the evolution of the reservoir. Our approach is validated through numerical simulations concerning the approximation of nonlinear functionals [47] and the prediction of chaotic time series [48].

Moreover, we investigate the effective utility of the quan-

[*] The two authors equally contributed to this work.
[†] e-mail: enrico.prati@unimi.it

tum reservoir by showing the role of quantum correlations in the memory capacity of the network, in particular when dealing with encoding ansatzs characterized by different entangling gates [22]. Indeed, the dynamics of the system subject to mid-circuit measurements is described by quantum trajectories that encode an input-dependent evolution, allowing the reservoir to retain information about past inputs without the need for a classical memory buffer.
Eventually, we test the algorithm on a currently available IBM superconducting quantum computer. It demonstrates improvements in the ability to memorize and process information over long time intervals compared to previous approaches based on intrinsic noise [37, 41], despite being implemented on newer, and thus less noisy, hardware. In conclusion, these results outline a robust strategy for scalable, noise-independent quantum reservoir computing that is compatible with fault-tolerant quantum devices.

RESULTS

In this Section, we present the main results of our work. Before that, we begin by summarizing the main concepts of gate-based quantum reservoir computing by distinguishing three possible approaches, which differ in terms of the strategy adopted to ensure memory retention. Next, we first demonstrate that induced dissipation compensates for quantum information loss, thus ensuring the persistence of the computational process. Then, we discuss the role of entanglement in the input encoding circuit by investigating the behavior of two- and three-qubit gates. Building on this a priori investigation, we verify the computational performance of the algorithm on two standard benchmarks. The first addresses the reconstruction of a nonlinear mapping, while the second aims at predicting a chaotic dynamical system. Eventually, we test our algorithm on IBM superconducting quantum computers, showing that the controlled damping algorithm allows better performance compared to implementations that rely only on intrinsic noise [37, 41].

Gate-based quantum reservoir computing

The general aim of sequential data processing with quantum reservoir computing consists in defining a mapping from an input space, constituted by temporal data $x \in \mathcal{I}$, into another target time series $y \in \mathcal{O}$, through read-out operations of the information encoded in the reservoir. Generically, denoting the state of the reservoir with $\rho(t)$, defined on the Hilbert space $\mathcal{H}$ that describes a quantum system, the evolution of a quantum reservoir computer can be written as

$$\begin{cases} \rho_t = \mathcal{T}(\rho_{t-1}, x_t) \\ y_t = h(\rho_t) \end{cases} . \tag{1}$$

where the time evolution of the system over discretized time intervals is expressed as $\ldots \rho_{t-1}, \rho_t, \rho_{t+1} \ldots$, $\mathcal{T} : \mathcal{S}(\mathcal{H}) \to \mathcal{S}(\mathcal{H})$ is any completely positive trace preserving (CPTP) channel that describes the dynamics of the quantum system, and h is a suitable readout operation. Then the reservoir computer defines a causal functional $\mathcal{C} \colon \mathcal{I} \to \mathcal{O}$, which acts as

$$y_t = \mathcal{C}(x)_t = \mathcal{C}\left(x_{|_t}\right) = h\left(\mathcal{T}(\rho_{t-1}, x_t)\right) \tag{2}$$

where $x_{|_t} = (x_0, \ldots, x_{t-1}, x_t)$ indicate the input sequence truncated at time t. In particular, the training of a reservoir computer consists of finding the readout weights, encoded in the readout operation h, in order to minimize the discrepancy between the target functional $S(x)$ and the functional $\mathcal{C}$ induced by the dynamics. In principle, any quantum system may act as the reservoir, as long as the sequential input data can be effectively embodied in the evolution of the system and the reservoir dynamics adequately preserve the memory of past inputs. In particular, while the current state of the system must depend in a non-trivial way on the past values of the time series, it must also be able to give greater importance to more recent inputs progressively, thus achieving the so-called fading memory property [49].

Different paradigms for the implementation of quantum reservoir computing

We now briefly review the three main approaches to gate-based quantum reservoir computing, which are schematically summarized in Fig. 1. The first uses classical statistics over mixed states with an engineered reset to balance memory [34, 35]. In particular, the input data is s encoded by randomly applying one of two fixed unitaries U_1 and U_2, with classical probabilities depending on the input value. A second approach is introduced in Refs. [38, 39], where the input information is encoded directly into the quantum state of a register of accessible qubits via a parametric unitary. After the encoding stage, the accessible qubits interact with a hidden reservoir—responsible for storing the past information processed by the network—through a fixed unitary operation, and are subsequently measured to extract the output data. In this approach, the non-unital character of the dynamics is ensured by the repeated reset of the accessible qubits. In the third approach, firstly introduced in Ref. [37], — also pursued in this work — operates without resets of the accessible qubits, leveraging quantum trajectories and controlled damping to maintain memory in fully accessible systems. In particular, the input is again encoded through parametric unitaries acting on the accessible qubits, after which the entire register is measured and allowed to evolve from the corresponding post-measurement collapsed state. Repeated measurements of the quantum circuit pose an issue, since it is well known that they progressively induce information loss, eventually hindering the computational capabilities

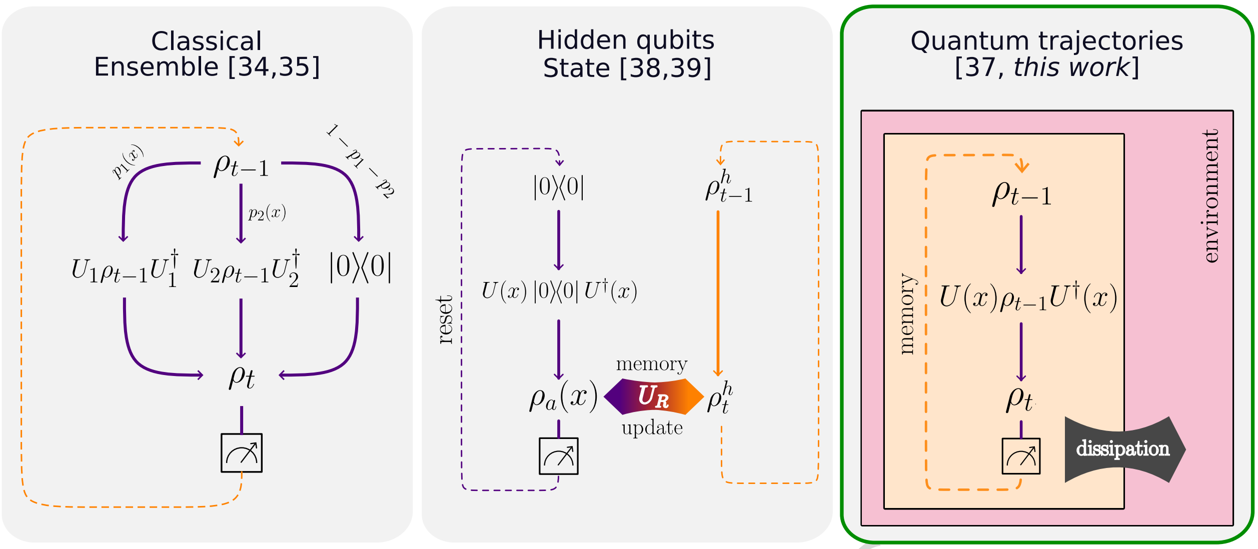

FIG. 1

of the reservoir computer. As shown in these works, the information loss induced by repeated mid-circuit measurements must be counteracted by introducing an appropriate dissipation mechanism capable of ensuring the non-unitality of the resulting evolution. Our work shows how the controlled damping of the qubits can overcome loss of coherence, making the algorithm stable over time intervals significantly longer than the typical information survival time in the measured system.

Information retention by induced dissipation

We begin the presentation of our results by demonstrating how our algorithm prevents information loss during input processing, enabling sustained reservoir computing. In many gate-based quantum reservoir computing implementations [37, 38, 41], including the one presented in this work, the input data are sequentially injected into the qubits register, and, concurrently, decoded by repeated mid-circuit measurements on the computational basis. Therefore, given that no reset of the accessible qubits is performed, the state of the system is described by a mixture that encompasses all possible trajectories depending on the measurement outcomes, when considered as a statistical ensemble of many repetitions of the circuit. A key requirement for effective reservoir computing is that the system's dynamics is sufficiently expressive to reliably separate distinct inputs [50], such that the resulting states $\rho_x = \mathcal{U}_x\rho$ and $\rho_y = \mathcal{U}_y\rho$ are distinguishable after the input injection through some unitary matrix. Following Holevo–Helstrom theorem [51, 52], the distinguishability of two different quantum states is quantified by their trace distance, namely $\mathcal{D}(\rho,\sigma) = \frac{1}{2}\|\rho-\sigma\|_1$ with $\|A\|_1 = \mathrm{Tr}\left(\sqrt{A\,A^\dagger}\right)$. In particular, as discussed in Ref. [41], the difference in trace norm of two internal states of the reservoir, corresponding to distinct values of the input, respectively x and y, is related to the distance between the internal state of the reservoir ρ before input injection and the maximally mixed state $\hat{\mathcal{I}}/d$, namely

$$\mathcal{D}\left(\rho_x-\rho_y\right) \leq C(U,|x-y|)\cdot\mathcal{D}\left(\rho-\hat{I}/d\right). \qquad (3)$$

Here is the maximally mixed state, with $d = 2^N$ denoting the dimension of the related Hilbert space. The constant C depends on the unitary used for the encoding and the difference between the inputs. In particular, it can be computed explicitly by exploiting the continuous dependence of unitary matrices on their parameters and the linearity of unitary evolution. The two constants, denoted with $C(U)$ and $C(|x-y|)$, depend on the unitary used for the input injection and the distance input values, respectively. In particular, a larger deviation from $\hat{I}$ improves the separability of distinct inputs [41]. As shown in Fig. 2, the repeated mid-circuit measurements drive the internal state of the reservoir ρ_t in the maximally mixed state, $\hat{I}/d$, in which the encoding of any input value is ineffective since $\mathcal{U}_x\hat{I} = \hat{I}$ for any value x. Indeed, any unitary rotation does not increase the distance from the identity, since

$$\left\|\mathcal{U}_x\rho-\hat{I}/d\right\|_1 = \left\|\mathcal{U}_x(\rho-\hat{I}/d)\right\|_1 = \left\|\rho-\hat{I}/d\right\|_1 \qquad (4)$$

while projective measurements decrease it, namely

$$\left\|\Pi\rho\Pi^\dagger-\hat{I}/d\right\|_1 = \left\|\Pi(\rho-\hat{I}/d)\Pi^\dagger\right\|_1 \leq \left\|\rho-\hat{I}/d\right\|_1. \qquad (5)$$

In particular, Equation (4) follows from the linearity of unitary transformation, while (5) follows from the triangular inequality for projectors. We refer to the Supplementary Material of [38] for further details. Such

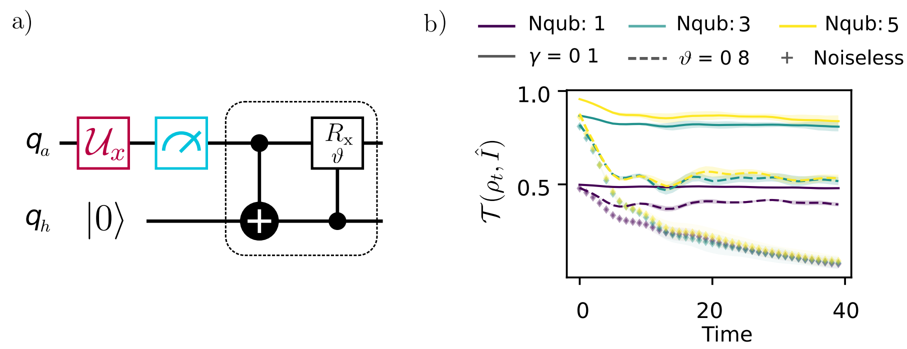

FIG. 2

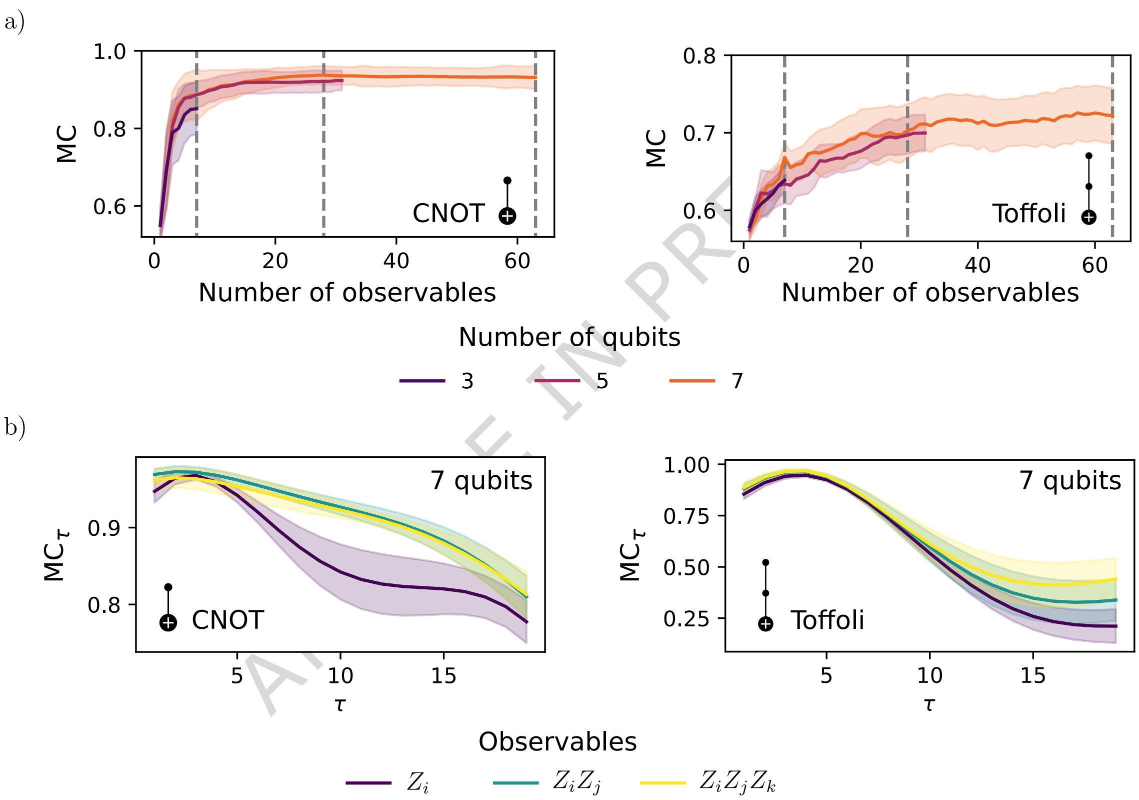

FIG. 3

concentration of the reservoir in the maximally mixed state can be prevented by exploiting the action of a non-unital channel. We recall that a quantum channel $\mathcal{E}$ is non-unital when $\mathcal{E}(\hat{I}) = \hat{I}$. As an example, the amplitude damping channel can be adopted for this purpose, as it tends to relax the qubit state toward $|0\rangle$ with a decay rate γ [37, 41, 42]. We recall that amplitude damping can be easily modeled by the application of the quantum channel described by the Kraus operators $K_0 = \begin{pmatrix} 1 & 0 \\ 0 & \sqrt{1-\gamma} \end{pmatrix}$, $K_1 = \begin{pmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{pmatrix}$. Although this approach offers significant advantages—being implementable on quantum computers with high dissipation rates—it may fail to deliver good performance, as it is hard to properly tune the noise using error mitigation techniques. While in principle it may be possible to enhance dissipation in many practical implementations of quantum computers, such as photonic or superconducting devices, it is usually difficult to design error mitigation techniques that specifically correct only the unital part of the noise, such as depolarizing and dephasing.

Alternatively, we propose a new approach by exploiting artificial dissipation toward an ancilla register to remove part of the superposition of the circuit. Ancilla-assisted digital simulations of open-system dynamics on gate-based quantum processors have been previously pro-

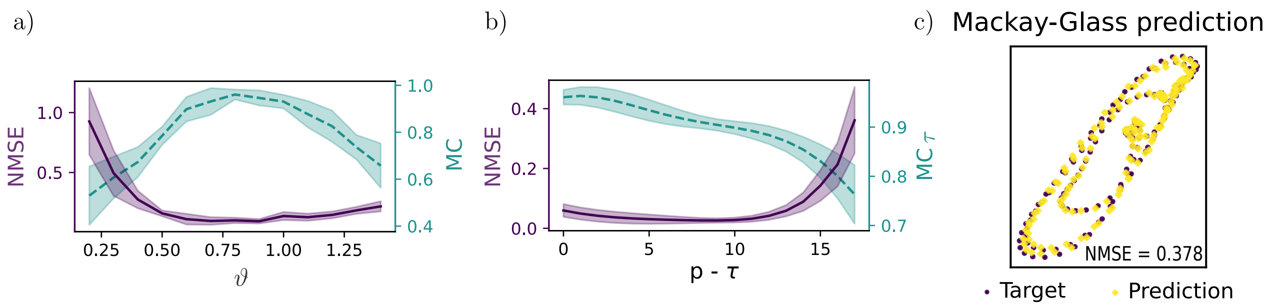

FIG. 4

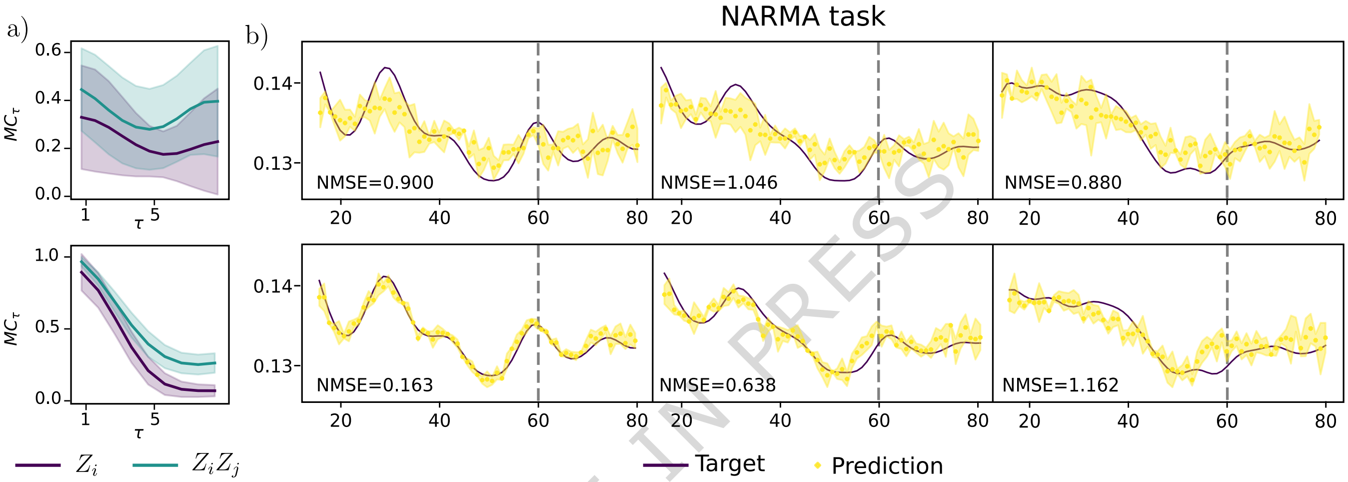

FIG. 5

posed and demonstrated, including implementations on IBM devices [53–55]. The methodology presented in this work proposes to induce damping through controlled rotations of ancilla qubits, thereby resulting in an amplitude amplification of the zero state of the reservoir qubits. This mechanism prevents the loss of quantum information and steers the state away from the maximally mixed state. Compared to external dissipation, the digital approach allows to perform the optimal tuning of the parameter that controls the dissipation. In this work, we induce the damping by connecting each qubit to an ancilla in the state $|0\rangle$ and a controlled rotation $CR_X(\vartheta)$ of a fixed angle ϑ, then partially damping the state of the qubit as

$$\alpha\,|0\rangle + \beta\,|1\rangle \underset{CR_X(\vartheta)}{\rightsquigarrow} (\alpha + i\sin(\vartheta/2))\,|0\rangle + \beta\cos(\vartheta/2)\,|1\rangle \tag{6}$$

after the input injection, as described in detail in Fig. 2. This setup provides a minimal example of how to engineer non-unital dynamics. Since all gates implemented on a quantum computer are unitary—and therefore unital, preserving the identity operator—it is necessary to introduce at least one additional qubit acting as an effective environment into which the system can dissipate. In our approach, non-unital behavior in the reservoir is enabled by resetting the ancilla before each dissipation step, thereby reinitializing it to a fixed state $|0\rangle$. As a reset is a non-unitary operation, it allows the reduced dynamics of the reservoir to become non-unital. Ideally, one should employ a rotation directly controlled on the ancilla being in the $|0\rangle$ state, which would yield the most compact circuit. However, since such a gate is typically not available as a native primitive, we implement the control using a CNOT gate followed by a controlled rotation, at the cost of additional circuit depth. We show in Fig. 2 that both methods, namely dissipation due to amplitude damping and that towards an ancilla, ensure that the internal state of the reservoir is kept away from the maximally mixed state, thereby guaranteeing the long-term operation of the algorithm, in contrast to what is observed in previous works [37, 56]. Indeed, the stability of the internal state of the reservoir in terms of its purity allows for an a-priori infinitely long computation.

Input encoding and correlated observables

After establishing an algorithm for the theoretical functioning of a gate-based echo state network, we now pass to investigate the effect of different encoding unitaries on its memory capacity. In particular, in this Section, we test two different input encoding ansatz, involving two-qubit and three-qubit logic gates, respectively. We show that, correspondingly, leveraging observables involving multiple qubits enhances the memory capacity of the reservoir. Indeed, adding entangling gates in the input encoding unitaries allows the system to exploit more computational degrees of freedom. As a proof of principle, we test two sets of unitaries, including interactions of two and three qubits, respectively, in addition to rotations that encode the current input value. In the first case, the input encoding $\mathcal{U}_x$ is composed of CNOT gates with bond dimension 1 and rotations $R_X(x)$. On the other hand, three-qubit entangling gates are embodied by Toffoli gates CXX. In particular, the angles of the rotations are exploited for the encoding of the value of the input. The precise ansatz of the two circuits is detailed in the Methods section. The reservoir signal is reconstructed from the expectation values of the Pauli-Z observables. In addition to the individual observables Z_i, $i = 1, \ldots, N_a$, we include the expectation values of joint observables involving multiple qubits, such as $Z_i \otimes Z_j$, $Z_i \otimes Z_j \otimes Z_k, \ldots$, and so on. Notably, this does not require additional measurements of the system, since all of these observables commute. Yet, because joint observables are sensitive to correlations across multiple qubits, they enable the discrimination of entangled states that would otherwise be indistinguishable. This, in turn, truly allows for the exploitation of the system's quantum degrees of freedom, thus enforcing a potential quantum advantage in the reservoir computation. This point is clarified by a trivial example involving the separable two-qubit state $|{+}{+}\rangle$ and an entangled one, $|\phi^+\rangle$. Indeed, while they are indistinguishable employing single observables, since $\langle Z_i\rangle_{|++\rangle} = \langle Z_i\rangle_{|\phi^+\rangle} = 0$ for $i = 1, 2$, they are discriminated by joint observables, since $\langle Z_1 \otimes Z_2\rangle_{|++\rangle} = 0$ but $\langle Z_1 \otimes Z_2\rangle_{|\phi^+\rangle} = 1$. We tested the effect of joint observables on the memory capacity of the reservoir. In this test, as detailed in the Methods, the reservoir computer is trained via linear regression on the readout weights to reproduce a sequential input x_t delayed in time by a fixed window τ, namely $(\hat{y}^\tau)_t = x_{t-\tau}$. Then, the memory capacity (MC) is quantified by the amount of variance in the delayed input $\hat{y}^\tau$ that can be recovered from the trained output y^τ [11], namely

$$\mathrm{MC}_\tau = \frac{\mathrm{Cov}\,(y^\tau, \hat{y}^\tau)}{\mathrm{Var}(y^\tau)\mathrm{Var}(\hat{y}^\tau)}\,. \tag{7}$$

Moreover, we quantify the overall memory capacity as

$$\mathrm{MC} = \frac{1}{\tau_{\max}} \sum_{\tau=0}^{\tau_{\max}} \mathrm{MC}_\tau. \tag{8}$$

As expected, in the case of input encoding that uses only two-qubit gates, only the exploitation of two-qubit observables in the readout of the reservoir's information significantly improves the memory capacity. On the other hand, observables that correlate three qubits provide benefits only in the case of input encoding with Toffoli gates. We summarize these results in Fig. 3, where we show the memory capacity as a function of the number of Z observables considered in the construction of the readout. The experiment is repeated for 12 different input instances of length $T = 200$ and three reservoir circuits consisting of 3, 5, and 7 qubits, respectively. Noticeably, the circuit based on CNOT gates performs better than the one relying on the Toffoli gate. This can be understood by observing that, as shown in the Methods section in Fig. 6, the circuit layout allows the inclusion of one less Toffoli gate compared to the CNOT case. This allows for fewer interactions between qubits, which may limit the complexity of the effective dynamics that can be generated within the reservoir. We expect that this difference is particularly relevant when working with a reduced number of qubits.

Performance on benchmark tasks

Building on previous insights, we test our algorithm on two benchmark tasks for reservoir computing. First, we investigate the memory capacity of the quantum echo state network by reconstructing the NARMA-p functional [47]. Moreover, we demonstrate the predictive capabilities of the network by forecasting the next step in the dynamics of a Mackey-Glass attractor in its chaotic regime. Both tasks are detailed in the Methods section. First, using a local simulator, we selected the optimal damping rate for the task under consideration. As a typical issue in reservoir computing, in which a strategy for a a-priori tuning of the hyperparameter often lacks, the best damping rate depends on both the task and the specific features of the circuit, and, currently, there is no general method to determine the optimal damping rate a priori. However, the network performs well on a sufficiently generic reconstruction task such as NARMA, precisely in the regime where it also exhibits the highest memory capacity. Thus, memory capacity serves as a reliable indicator of the network's performance. The results are reported in panels 1–2 of Fig. 4. Each experiment was repeated 12 times using a 7-qubit circuit. Implementation details can be found in the Methods section. We refer to the Supplementary material for an in-depth analysis of the NARMA task, which allows for relating the absolute value of the NMSE reported in the paper to the actual accuracy in reconstructing the sequence. Moreover, as further evidence of the effectiveness of our algorithm, we employed the echo state network for the forecasting of chaotic dynamics. In this task, the reservoir computer receives at each step an input value taken from a time series governed by a differential equation and is required to output the next step in the sequence. The results of the experiment are reported in the panel c in Fig. 4.

Experiment on a superconducting quantum computer

Finally, we validate our algorithm by conducting experiments on a cloud-accessible superconducting quantum computer. Although these experiments are hindered by some limitations imposed by the remote computing service, such as the maximum number of gates executable in a single run, they show positive outcomes that improve the state of the art. Indeed, similar experiments were already conducted in previous papers [37, 41], achieving reasonable results for NARMA-2. Interestingly, these results were obtained employing older quantum hardware characterized by greater damping, which naturally enables the computation even in the absence of a specific algorithm for the amplitude amplification theoretically required along with repeated measurements. In this work, our algorithm effectively enables the execution of tasks requiring greater network memory, obtaining satisfactory results up to NARMA-5, while for larger memory intervals, the results progressively deteriorate. However, motivated by tests performed on an emulator that does not suffer from the limitations inherent to quantum hardware, we are confident that our algorithm allows the execution of tasks requiring the integration of information over longer periods. As shown in Fig. 5, the induced damping drastically improves the memory capacity of the reservoir computer. Moreover, the observation on the beneficial effect of considering correlated observables along with two-qubit entangling gates is confirmed. For this reason, in Fig. 5 we report results of the training that employ the single observables Z_i and the correlated double observables $Z_i \otimes Z_j$. In these experiments, we employ only CNOT gates in the input encoding unitary to better accomplish the limitation imposed by the remote service in terms of entangling gates, allowing for the execution of a reasonable number of shots of the circuit. The application of induced damping improves the network's performance, allowing the execution of the NARMA-p task with good results up to $p = 5$. For harder tasks, the results begin to deteriorate due to quantum noise in the hardware. Indeed, while non-unitary noise such as amplitude damping enhances performance by contributing to the learning [37, 41, 42], other noise sources like dephasing and depolarizing progressively degrade the qubit state, limiting the memory capacity of the reservoir computer.

DISCUSSION

We implemented a gate-based echo state network utilizing a qubit circuit for processing sequential data, proposing an algorithm that ensures the effectiveness of the dynamics of the quantum reservoir. Indeed, damping has already been recognized as a necessary resource for enabling the functioning of a gate-based echo state network [37], as it ensures the separability of inputs over long-term computations [41]. In particular, the loss of information due to repeated mid-circuit measurements is prevented by inducing the damping of the accessible qubits. In this work, we proposed a ad-hoc ansatz involving controlled rotations that, in contrast with previous approaches based on intrinsic noise in the quantum hardware, ensures the full controllability of the damping rate. Our algorithm enables information processing over timescales significantly exceeding the qubit coherence times. Nevertheless, its performance remains constrained by the capabilities of the underlying hardware, in particular by the presence of unital noise such as depolarizing and dephasing processes, as well as by the limited measurement fidelity. Moreover, the current implementation requires a non-negligible ancilla overhead, which can be partially mitigated through ad hoc reset protocols for the ancilla qubits or by employing multiple ancilla registers. Notably, it does not require the reset of the accessible register [38]. Moreover, unlike previous approaches that rely on the intrinsic noise of the hardware to ensure non-unital dynamics [37, 41], the explicit control of the induced damping allows for optimizing the dissipation-induced fading memory according to the nature of the task at hand. Furthermore, building up on our algorithm, we highlight the computational utility of the degrees of freedom associated with the state space of the quantum reservoir, which are fully exploited by encoding the data into entangled states and subsequently using correlated observables for the readout of the information processed by the reservoir, which, in turn, improves the network's memory capabilities. Eventually, we test the method on standard benchmarks of reservoir computing, numerically and with experiments on a cloud-accessible superconducting quantum hardware. These experiments demonstrate the feasibility of our algorithm on currently available quantum hardware by improving the state of the art in terms of the memory capacity and the expressiveness of the echo state network in reconstructing nonlinear functionals that require a longer integration time. Moreover, our algorithm is well-suited for future implementation on noiseless quantum computers, since it does not rely on any built-in hardware noise [37, 41], but instead exploits a controlled dissipative mechanism that can be tailored and optimized according to the specific task under consideration.

METHODS

Architecture of the gate-based echo state network

We begin by describing the architecture of our gate-based echo state network, which employs an N-qubit circuit to encode and process time-dependent information. Then, a linear readout is trained by linear regession to replicate a mapping $S(u) = \hat{y}$ between an input sequence $u = \{u_t\}_{t=0,\ldots,L}$ and a target sequence $\hat{y} = \{\hat{y}_t\}_{t=0,\ldots,L}$. We denote with ρ_t the density operator that describes the state of the N-qubit circuit at time t. Its one-step evolution $\widetilde{\rho}_{t+1} = \mathcal{U}_{x_{t+1}}(\rho_t)$ is given by a unitary evolution, which encodes the value of the current input x_{t+1} in the angles of some rotations. Precisely, the two-qubit entangling circuit evolves under the application of the following gates, namely

$$U(x) = \prod_{i=0}^{N_a-1} \overline{U}_{i,i+1}\,, \\ \overline{U}_{i,i+1} = \mathrm{CX}^{i,i+i}\left(\hat{I}^0 \otimes \hat{I}^1 \dots R_X^i(x) \otimes \dots \otimes Id^{N_a}\right) \tag{9}$$

where we are denoting with $\mathrm{CX}^{i,i+i}$ the CNOT gate with i^{th} and $i+1^{\mathrm{th}}$ qubits of the circuit as control and target, respectively, and with $R_X^i(x)$ the rotation of the i^{th} qubit with angle x. On the other hand, the ansatz characterized by 3-qubit gates is defined as

$$U(x) = \prod_{i=0}^{N_a-2} \overline{U}_{i,i+1,i+2}\,, \\ \overline{U}^3_{i,i+1} = \mathrm{CCX}^{i,i+i,i+2}\left(\hat{I}^0 \otimes \hat{I}^1 \dots \otimes R_X^i(x) \otimes \dots \otimes Id^{N_a}\right) \tag{10}$$

where $\mathrm{CCX}^{i,i+1,i+2}$ is the Toffoli gate with qubit i and $i+1$ as control, and $i+2$ as target. After the input encoding, the mid-circuit measurement of the register is performed on the computational basis and, as opposed to other approaches [38, 39], the system is let evolve in the collapsed state after the measurements. In this way, at each time step, the ensemble ρ_t, which describes the state of the circuit when considering the experiment repeated for many shots, is reminiscent of the sequential input $x_{t'}$ for any $t' \leq t$. Formally, we have

$$\rho_{t+1} = \Pi_N \widetilde{\rho}_{t+1} \Pi_N^\dagger \tag{11}$$

where, denoting with $m_{s,t+1} \in \{0, 1\}$ the outcome of the measure of the i^{th} qubits i at time $t+1$, we are writing

$$\Pi_N = \bigotimes_{i=1}^{N} |m_{i,t+1}\rangle\langle m_{i,t+1}| \tag{12}$$

Concurrently, the reservoir signal at time t is constructed online as the expectation value of the Z_i Pauli operators and their correlated tensor products, namely

$$z_t = \left[\langle Z_1\rangle_{\rho_t}, \dots, \langle Z_M\rangle_{\rho_t}, \dots, \langle Z_i \otimes Z_j \otimes Z_k\rangle_{\rho_t} \dots\right]^T. \tag{13}$$

In the following Section, we first describe in detail the ansatz of the circuit, also described in Fig. 6, and the training procedure that allows for reconstructing the wanted functional given the reservoir signal.

The architecture of the two encoding circuits

The two circuits employed in the experiment, which differ in the entangling gates in the input encoding, with CNOT and Toffoli gates, respectively, are reported in Fig. 6. After the input encoding, the accessible qubits are measured online, and the system is left to evolve in the collapsed state. We remark that mid-circuit measurements are now available on many of the quantum hardware platforms currently available, and represent a key feature for the development of many quantum algorithms. After the measurements, the partial damping, driven by controlled rotation with a fixed parameter ϑ, is induced by connecting the reservoir with an ancilla register. Although in principle the dissipation would be effective at any stage of the evolution, we choose to apply it only after each measurement. Indeed, this ensures maximal separability of the inputs at the time of information extraction, since the damping tends to shrink the distance between different quantum states. The same procedure is repeated for each value of the input sequential data. While the computation on a simulator benefits from the repeated reset of a single qubit, which essentially requires the multiplication of smaller matrices, in the case of implementation on quantum hardware, it is possible to leverage all available qubits to reduce the number of reset operations. This allows for a reduction in the computational time required for each reset and decreases operational error.

Training of linear readout

We describe now the training procedure, which essentially reduces to a linear regression of the reservoir signal on the desired task. Indeed, reservoir computing aims to best reproduce a given nonlinear map $S(u) = \hat{y}$ by training a linear readout W. Namely, we aim to compute the optimal weights so that

$$W z_t = y_t \simeq \hat{y}_t\,. \tag{14}$$

After an initial washout period T_{wo}, required to erase the dependence on the initial state of the network and fulfill the so-called echo state property [11], the reservoir signal is stored in the matrix $H = \{z_t\}_{t=T_{\mathrm{wo}}+1,\dots,T_{\mathrm{tr}}}$, where we are denoting with $(T_{\mathrm{wo}}, T_{\mathrm{tr}})$ the time label of the training set of data. Then, the optimal readout weights, namely the ones for that $y_t = W z_t$ minimize the distance $\|\hat{y} - y\|$ are computed by linear regression. Precisely, they are computed exploiting the Moore-Penrose pseudoinverse matrix, namely $w = (H^T H)^{-1} H^T \hat{y}_{\mathrm{tr}}$, where $\hat{y}_{\mathrm{tr}} = \{\hat{y}_t\}_{t=T_{\mathrm{wo}}+1,\dots,T_{\mathrm{tr}}}$ is the portion of target sequence used for the training. We refer to Ref. [35] for further details on the training algorithm. After the training, the predicted values of y are given by

$$y_t = z_t \cdot W \qquad \text{for any } t \in [T_{tr}+1, L]\,. \tag{15}$$

Interestingly, the reservoir can be trained to fulfill different tasks that share the same input, since the internal dynamics is independent from the training algorithm on a specific task, thus reducing the computational cost

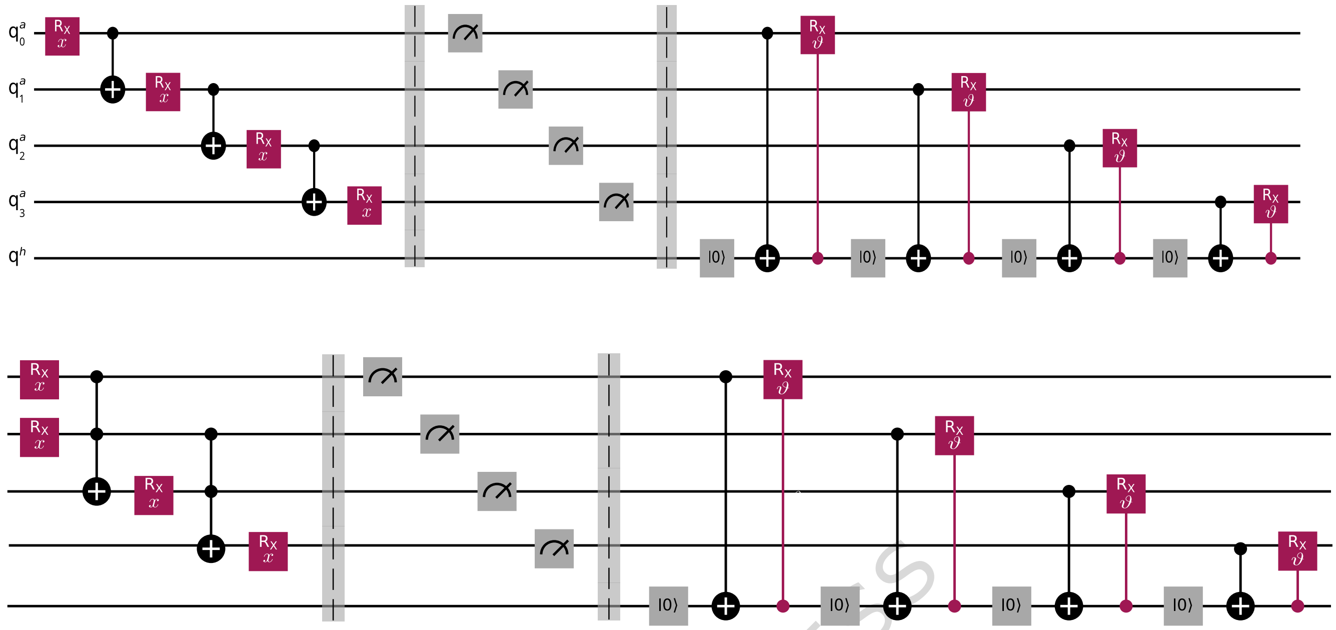

FIG. 6

of the execution. Moreover, as explained in the results, this allows for comparison of the intrinsic memory of the network – with the memory capacity task – and the performance on benchmark nonlinear tasks, as the NARMA task. To avoid overfitting in the training of the linear readout, we work in an overdetermined or underdetermined regime, where the length of the training set L is much larger or much smaller than the number of trainable parameters, i.e., the number of observables N_{obs}. In addition, the linear regression can be stabilized by introducing a regularization term, which improves the conditioning of the problem and further suppresses overfitting. Throughout the training, we fix the regularization parameter to $\alpha = 0.001$. We discuss further details on the problem of overfitting in the Supplementary material.

Benchmark tasks

Here, we formally describe the two tasks employed in the experiments to validate the algorithm. They represent two typical benchmarks in reservoir computing implementations. In particular, the first evaluates the network's ability to reconstruct past inputs, while the second assesses its predictive capabilities of a chaotic dynamical system. The Nonlinear Auto-Regressive Moving Average dynamics (NARMA) task requires reproducing a nonlinear functional with past dependence in the sequential input, testing how well a reservoir computer can handle progressively longer dependencies. Formally, we define the NARMA-p functional as

$$\hat{y}_t = \alpha\, y_{t-1} + \beta \left(\sum_{l=0}^{p-1} y_{t-l}\right) + \gamma\, u_{t-p+1} u_t + \delta\,. \tag{16}$$

with $\alpha, \beta, \gamma, \delta = (0.3, 0.05, 1.5, 0.1)$. The input is sampled from the periodic function

$$u_t = 0.1 \sin\left(2\pi \frac{2.11t}{T}\right) \sin\left(2\pi \frac{3.73t}{T}\right) \sin\left(2\pi \frac{4.11t}{T}\right). \tag{17}$$

In the work, we fix $T = 200$ and we employ an input sequence with length 200 and 80 for the experiment on the local simulator and the remote quantum hardware, respectively. The test set has lengths of 130 and 50, while the first 20 steps are used for the washout of the initial condition of the reservoir. This choice is required to fulfill the computational limitations of the remote quantum service made available by IBM. We run simulations with 50.000 shots, which represents a valid regime for approximating the expected values deterministically. The experiments on quantum hardware involve 10800 repetitions of the circuit, divided into 120 batches with 60 shots each.

We test the predictive capabilities of the network by forecasting the next-step evolution of the Mackey-Glass equation in the chaotic regime. In the Mackey-Glass model, the dynamics is driven by a time-delay differential equation that exhibits chaotic behavior for certain values of the parameter τ. Formally, the equation reads

$$\frac{dx}{dt} = \beta \frac{x(t-\tau)}{1 + x(t-\tau)^n} - \gamma x(t), \tag{18}$$

where we fix $\beta = 0.2$ and $\gamma = 0.1$ the positive constants that control the growth and decay rates, respectively, and τ is the delay parameter. For small values of τ, the system exhibits stable fixed points or periodic oscillations. For increasing values of τ, the system exhibits a transition into chaotic dynamics. Here, we fix $\tau = 17$

in the experiments, well within the range of the chaotic regime. Then, the task is defined as the next-step prediction, namely

$$\hat{y}_t = x(t+1)\,, \tag{19}$$

In the experiment, we consider an input sequence of length 2000, sampled from the continuous time series with a fixed period of length 10^{-2}. In both cases, the accuracy of the prediction is quantified by the normalized mean square error (NMSE), expressed as

$$\mathrm{NMSE}(y, \hat{y}) = \frac{\sum_{t=T_{\mathrm{tr}+1}}^{t=L} (\hat{y}_t - y_t)^2}{\sum_{t=T_{\mathrm{tr}+1}}^{t=L} (\hat{y}_t - \mu)^2} \tag{20}$$

where y is the vector of the predicted values after training, $\hat{y}$ is the target sequence, and $\mu = L - T_{\mathrm{tr}} \sum_{t=T_{\mathrm{tr}+1}}^{t=L} \hat{y}_t$ is the mean value of the target sequence. The training and the test set have lengths of 1300 and 500 steps, respectively.

Experimental settings

Classical simulations were conducted on a 36-core Intel Xeon workstation (4.3 GHz) with 128 GB of RAM.
The quantum operations were executed on the *ibm_torino* QPU, a Heron $r1$ superconducting processor that features 133 qubits arranged in a heavy-hexagonal lattice architecture [57]. This device utilizes tunable couplers to dynamically modulate interaction strengths, effectively suppressing residual crosstalk that might otherwise mask delicate correlation effects. At the time of execution, the QPU reported a median two-qubit error rate of 2.5×10^{-3}, with median T_1 and T_2 coherence times of $190\,\mu$s and $141.7\,\mu$s, respectively. Further calibration data and hardware specifications are detailed in Ref. [58]. A distinctive feature of our experimental protocol is the omission of quantum error mitigation (QEM) or correction. This was a deliberate design choice aimed at exploiting the intrinsic hardware noise as a computational resource for the quantum reservoir, acknowledging the role of non-unital noise in enhancing the reservoir's overall performance. Other noise channels do not significantly degrade the reservoir's expressive power, hence there is no evident need to mitigate them. This work establishes a baseline for the performance of unmitigated quantum reservoirs, leaving the engineering of specialized noise-aware error mitigation for future developments.

DATA AVAILABILITY

The datasets generated and/or analyzed during the current study are not publicly available due to technical limitations and their use in ongoing related research, but are available from the corresponding author on reasonable request.

ACKNOWLEDGEMENTS

E.R. and E.P. are supported by Qxtreme project funded by the Partenariato Esteso FAIR (grant N. J33C22002830006). F.M. and E.P. are supported by PRIN-PNRR PhysiComp (grant N. G53D23006710001). L.N. is partially funded by ENI S.p.A.

AUTHORS CONTRIBUTION

E.R. and F.M. designed the algorithm, the computational framework and conceived the experiments. E.R E.R, F.M. and L.N. analysed the data and curated the visualization. E.R. and F.M. wrote the first of the manuscript with input from all authors. All the authors contributed to the final version of the manuscript. E.P. conceived the study and were in charge of overall direction and planning. E.R. and F.M. equally contributed to this work.

COMPETING INTERESTS

Authors declares no competing interests.

REFERENCES

[1] Deutsch, I. H. Harnessing the power of the second quantum revolution. *PRX Quantum* **1**, 020101 (2020).
[2] Bharti, K. *et al.* Noisy intermediate-scale quantum algorithms. *Rev. Mod. Phys.* 015004 (2022).
[3] Prati, E. Quantum neuromorphic hardware for quantum artificial intelligence. In *Journal of Physics: Conference Series*, vol. 880 (IOP Publishing, 2017).
[4] Schuld, M. & Petruccione, F. *Chapter in: "Machine learning with quantum computers"*, vol. 676 (Springer, 2021).
[5] Lloyd, S., Mohseni, M. & Rebentrost, P. Quantum principal component analysis. *Nature physics* **10**, 631–633 (2014).
[6] Simen, A. *et al.* Quenched quantum feature maps (2025). 2508.20975.
[7] Innocenti, L. *et al.* Potential and limitations of quantum extreme learning machines. *Communications Physics* **6**, 118 (2023).
[8] McClean, J., Boixo, S., Smelyanskiy, V., Babbush, R. & Neven, H. Barren plateaus in quantum neural network training landscapes. *Nature Communications* (2018).
[9] Larocca, M. *et al.* Barren plateaus in variational quantum computing. *Nature Reviews Physics* 1–16 (2025).
[10] Mujal, P. *et al.* Opportunities in quantum reservoir computing and extreme learning machines. *Advanced Quantum Technologies* **4** (2021).
[11] Jaeger, H. Short term memory in echo state networks (2001).
[12] Maass, W. W., Natschläger, T. & Markram, H. Real-time computing without stable states: a new framework

for neural computation based on perturbations. *Neural Computation* (2002).
[13] Maass, W. & Markram, H. On the computational power of circuits of spiking neurons. *J. Comput. Syst. Sci.* (2004).
[14] Nakajima, K. & Fischer, I. *Reservoir Computing. Theory, Physical Implementations, and Applications* (Springer Singapore, 2021).
[15] Monzani, F. & Prati, E. Universality conditions of unified classical and quantum reservoir computing. *Neurocomputing* 130391 (2025).
[16] Tanaka, G. *et al.* Recent advances in physical reservoir computing: a review. *Neural Networks* **115** (2019).
[17] Marković, D. & Grollier, J. Quantum neuromorphic computing. *Applied Physics Letters* **117** (2020).
[18] Senanian, A. *et al.* Microwave signal processing using an analog quantum reservoir computer. *Nature Communications* **15** (2024).
[19] Franceschetto, G., Płodzień, M., Lewenstein, M., Acín, A. & Mujal, P. Harnessing quantum back-action for time-series processing. arXiv:2411.03979 (2024).
[20] Ghosh, S., Opala, A., Matuszewski, M., Paterek, T. & Liew, T. C. H. Quantum reservoir processing. *npj Quantum Information* **5** (2019).
[21] Ghosh, S., Krisnanda, T., Paterek, T. & Liew, T. C. H. Realising and compressing quantum circuits with quantum reservoir computing. *Communications Physics* **4** (2021).
[22] Götting, N., Lohof, F. & Gies, C. Exploring quantumness in quantum reservoir computing. *Physical Review A* **108**, 052427 (2023).
[23] Nakajima, M., Tanaka, K. & Hashimoto, T. Scalable reservoir computing on coherent linear photonic processor. *Communications Physics* **4** (2021).
[24] García-Beni, J., Giorgi, G. L., Soriano, M. C. & Zambrini, R. Scalable photonic platform for real-time quantum reservoir computing. *Phys. Rev. Appl.* **20** (2023).
[25] Bravo, R. A., Najafi, K., Gao, X. & Yelin, S. F. Quantum reservoir computing using arrays of Rydberg atoms. *PRX Quantum* **3**, 030325 (2022).
[26] Govia, L. C. G., Ribeill, G. J., Rowlands, G. E., Krovi, H. K. & Ohki, T. A. Quantum reservoir computing with a single nonlinear oscillator. *Phys. Rev. Res.* **3**, 013077 (2021).
[27] Nokkala, J. *et al.* Gaussian states of continuous-variable quantum systems provide universal and versatile reservoir computing. *Communications Physics* **4** (2021).
[28] Spagnolo, M., Morris, J., Piacentini, S. *et al.* Experimental photonic quantum memristor. *Nature Photonics* **16**, 318–323 (2022).
[29] Selimović, M. *et al.* Experimental neuromorphic computing based on quantum memristor (2025). URL `https://arxiv.org/abs/2504.18694`. 2504.18694.
[30] Pfeffer, P., Heyder, F. & Schumacher, J. Hybrid quantum-classical reservoir computing of thermal convection flow. *Phys. Rev. Res.* **4** (2022).
[31] Ahmed, O., Tennie, F. & Magri, L. Prediction of chaotic dynamics and extreme events: A recurrence-free quantum reservoir computing approach. *Phys. Rev. Res.* **6** (2024).
[32] Dudas, J. *et al.* Quantum reservoir computing implementation on coherently coupled quantum oscillators. *npj Quantum Information* **9**, 64 (2023).
[33] Angelatos, G., Khan, S. A. & Türeci, H. E. Reservoir computing approach to quantum state measurement. *Phys. Rev. X* **11**, 041062 (2021).
[34] Chen, J., Nurdin, H. I. & Yamamoto, N. Temporal information processing on noisy quantum computers. *Phys. Rev. Appl.* **14** (2020).
[35] Molteni, R., Destri, C. & Prati, E. Optimization of the memory reset rate of a quantum echo-state network for time sequential tasks. *Phys. Lett. A* (2023).
[36] Suzuki, Y., Gao, Q., Pradel, K. C., Yasuoka, K. & Yamamoto, N. Natural quantum reservoir computing for temporal information processing. *Scientific reports* **12** (2022).
[37] Kubota, T. *et al.* Temporal information processing induced by quantum noise. *Physical Review Research* **5**, 023057 (2023).
[38] Hu, F. *et al.* Overcoming the coherence time barrier in quantum machine learning on temporal data. *Nature communications* **15**, 7491 (2024).
[39] Xiong, W. *et al.* On fundamental aspects of quantum extreme learning machines. *Quantum Machine Intelligence* **7**, 20 (2025).
[40] Martínez-Peña, R. & Ortega, J.-P. Quantum reservoir computing in finite dimensions. *Physical Review E* **107** (2022).
[41] Monzani, F., Ricci, E., Nigro, L. & Prati, E. Non-unital noise in a superconducting quantum computer as a computational resource for reservoir computing. arXiv:2409.07886 (2024).
[42] Domingo, L., Carlo, G. & Borondo, F. Taking advantage of noise in quantum reservoir computing. *Sci. Rep.* (2023).
[43] Sannia, A., Martínez-Peña, R., Soriano, M. C., Giorgi, G. L. & Zambrini, R. Dissipation as a resource for Quantum Reservoir Computing. *Quantum* **8**, 1291 (2024).
[44] Fefferman, B., Ghosh, S., Gullans, M., Kuroiwa, K. & Sharma, K. Effect of nonunital noise on random-circuit sampling. *PRX Quantum* **5** (2024).
[45] Mele, A. A. *et al.* Noise-induced shallow circuits and absence of barren plateaus (2024). 2403.13927.
[46] Noè, D., Rocutto, L., Moro, L. & Prati, E. Quantum parallel training of a Boltzmann machine on an adiabatic quantum computer. *Advanced Quantum Technologies* 2300330 (2024).
[47] Atiya, A. & Parlos, A. New results on recurrent network training: unifying the algorithms and accelerating convergence. *IEEE Transactions on Neural Networks* **11** (2000).
[48] Antonik, P., Gulina, M., Pauwels, J. & Massar, S. Using a reservoir computer to learn chaotic attractors, with applications to chaos synchronization and cryptography. *Phys. Rev. E* **98** (2018).
[49] Boyd, S. & Chua, L. Fading memory and the problem of approximating nonlinear operators with Volterra series. *IEEE Transactions on Circuits and Systems* **32**, 1150–1161 (1985).
[50] Martínez-Peña, R. & Ortega, J.-P. Input-dependence in quantum reservoir computing. *Physical Review E* **111**, 065306 (2025).
[51] Helstrom, C. W. Quantum detection and estimation theory. *Journal of Statistical Physics* **1**, 231–252 (1969).
[52] Holevo, A. S. Bounds for the quantity of information transmitted by a quantum communication channel. *Problemy Peredachi Informatsii* **9**, 3–11 (1973).

[53] García-Pérez, G., Rossi, M. A. C. & Maniscalco, S. Ibm q experience as a versatile experimental testbed for simulating open quantum systems. *npj Quantum Information* **6** (2020).

[54] Guo, Y.-M., Albarrán-Arriagada, F., Alaeian, H., Solano, E. & Alvarado Barrios, G. Quantum memristors with quantum computers. *Physical Review Applied* **18**, 024082 (2022).

[55] Saavedra-Pino, S., Inzulza, C., Román, P., Albarrán-Arriagada, F. & Retamal, J. C. Quantum simulation of entanglement dynamics in a quantum processor. *Physica Scripta* **99**, 095105 (2024).

[56] Yasuda, T. *et al.* Quantum reservoir computing with repeated measurements on superconducting devices (2023). 2310.06706.

[57] AbuGhanem, M. Ibm quantum computers: evolution, performance, and future directions. *The Journal of Supercomputing* **81** (2025).

[58] Miessen, A., Egger, D. J., Tavernelli, I. & Mazzola, G. Benchmarking digital quantum simulations above hundreds of qubits using quantum critical dynamics. *PRX Quantum* **5**, 040320 (2024).

Fig. 1. Three paradigms for gate-based quantum reservoir computing. Left: input encoding via the probabilistic application of fixed unitaries, with fading memory enforced by stochastic register resets [34, 35]. Center: input encoded through parametric unitaries acting on accessible qubits, which interact with a hidden reservoir before measurement; non-unitality is ensured by repeated resets of accessible qubits [38, 39]. Right: input encoding via parametric unitaries followed by repeated mid-circuit measurements of the full register, with information loss mitigated by suitable dissipative mechanisms [37, 41]. This latter approach is pursued in this work.

Fig. 2. Impact of damping on decoherence of the reservoir. a) The circuit for the induced dissipation after input encoding and mid-circuit measurements for 1 qubit. The dissipation is induced by a CNOT gate followed by a controlled rotation of angle ϑ. The same scheme applies to each qubit in the circuit. **b)** Trace distance of the circuit from the $\hat{I}$ along 40 time steps of input encoding and mid-circuit measurements, averaged over 10 repetitions with random inputs. If the dynamics is affected by amplitude damping (dashed line) or by induced dissipation towards an ancilla register (continuous line), the reservoir state remains deviated from $\hat{I}$ during the evolution, while in noiseless execution (crosses) it tends to the identity as predicted.

Fig. 3. Memory capacity with different encoding unitaries. a) Memory capacity with $\tau_{\max} = 20$ for increasing number of Z observables included in the readout, respectively for an input encoding with two and three-qubit entanglement. The dashed grey lines are for $N_{\text{obs}} = 7, 28, 63$ and correspond to the set of correlated observables involving 1,2, and 3 qubits, respectively, for a reservoir composed of 7 qubits. **b)** Memory capacity for increasing time-span (MC_τ) for the two different encoding anstatz. The three lines correspond to different readout observables. Purple line represents the caso of single ($N_{\text{obs}} = 7$) observables, while green and yellow lines correspond to adding double ($N_{\text{obs}} = 23$, and triple ($N_{\text{obs}} = 63$) correlated observables. The simulations employ an induced damping corresponding to $\vartheta = 0.8$.

Fig. 4. Memory capacity and benchmarks on a simulator. a) Memory capacity and mean error in solving the NARMA task for increasing value of the damping, parametrized by ϑ. Here, the NMSE is computed as the average of the NMSE for each NARMA-p task for $p \in (0, 20)$. **b)** Memory capacity and normalised error for NARMA task with increased time horizon. Namely, here NARMA is the mean error for each NARMA-p task with fixed p running on the x-axis. The damping is fixed at the observed optimal value $\theta = 0.8$. **c)** Prediction of the next step dynamics of the Mackay-Glass chaotic attractor. Only the test set is depicted. All the experiments in this Figure are conducted on a local simulator. In the Figure is reported the mean value of the error and an error band of 1 standard deviation.

FIG. 7: **Fig. 5. Experiments on a superconducting quantum computer. a)** Memory capacity for increasing time window without (above) and with (below) induced damping, corresponding to a fixed rotation angle $\vartheta = 0.6$. The solid line indicates the mean value over 5 executions, while the error band corresponds to 1 standard deviation. **b)** Reconstruction of the NARMA-p task for, from left to right, $p = 2, 5, 8$. The line above contains results for an execution without induced damping, while the line below reports the results for a damping corresponding to $\vartheta = 0.8$.The linear regression training was made by using the single observables Z_i and the correlated double observables $Z_i \otimes Z_j$. The experiment is run 5 times, and we represent the mean value of the prediction along with an interval of 1 standard deviation. The grey dashed line divides the training and test sets in the learning algorithm. The washout period is not depicted.

FIG. 8: **Fig. 6. Ansatz for a single time-step evolution.** The circuits employed as the reservoir computer with two and three-qubit entangling gates, respectively. This block is repeated for each of the processed time series, exploiting mid-circuit measurements during execution. The ancilla register q^h is employed for the induced dissipation.