Temporal Information Processing on Noisy Quantum Computers

Jiayin Chen,[1,2] Hendra I. Nurdin⬥,[1,*] and Naoki Yamamoto⬥[2,3]

[1] *School of Electrical Engineering and Telecommunications, The University of New South Wales (UNSW), Sydney, New South Wales 2052, Australia*

[2] *Quantum Computing Center, Keio University, Hiyoshi 3-14-1, Kohoku, Yokohama 223-8522, Japan*

[3] *Department of Applied Physics and Physico-Informatics, Keio University, Hiyoshi 3-14-1, Kohoku, Yokohama 223-8522, Japan*

The combination of machine learning and quantum computing has emerged as a promising approach for addressing previously untenable problems. Reservoir computing is an efficient learning paradigm that utilizes nonlinear dynamical systems for temporal information processing, i.e., processing of input sequences to produce output sequences. Here we propose quantum reservoir computing that harnesses complex dissipative quantum dynamics. Our class of quantum reservoirs is universal, in that any nonlinear fading memory map can be approximated arbitrarily closely and uniformly over all inputs by a quantum reservoir from this class. We describe a subclass of the universal class that is readily implementable using quantum gates native to current noisy gate-model quantum computers. Proof-of-principle experiments on remotely accessed cloud-based superconducting quantum computers demonstrate that small and noisy quantum reservoirs can tackle high-order nonlinear temporal tasks. Our theoretical and experimental results pave the path for attractive temporal processing applications of near-term gate-model quantum computers of increasing fidelity but without quantum error correction, signifying the potential of these devices for wider applications including neural modeling, speech recognition, and natural language processing, going beyond static classification and regression tasks.

I. INTRODUCTION

The ingenious use of quantum effects has led to a significant number of quantum machine learning algorithms that offer computational speedups [1,2]. While awaiting the demonstration of these quantum algorithms on full-fledge quantum computers equipped with quantum error correction, quantum computing has transitioned from theoretical ideas to the noisy intermediate-scale quantum (NISQ) technology era [3]. Hybrid quantum-classical algorithms using short-depth circuits are particularly suitable for implementation on NISQ devices. Many notable experimental demonstrations of NISQ devices employ hybrid algorithms for data classification [4] and quantum chemistry [5]. An on-going quest is to find interesting applications on quantum computers with increasingly lower noise profile, but not reaching a low enough threshold to enable continuous quantum error correction.

Here we propose a hybrid quantum-classical algorithm that *utilizes dissipative quantum dynamics* as reservoir computers (RCs) for temporal information processing on gate-model NISQ quantum computers. The goal for temporal information processing tasks, such as speech processing and natural language processing [6,7], is to learn the relationship between input sequences and output sequences. The RC framework uses an arbitrary but fixed dynamical system (in this case systems with dynamics described by state-space difference equations), the "reservoir," to map sequential inputs into its high-dimensional state space. Only a simple linear regression algorithm is required to optimize the parameters of a readout function to approximate target outputs. The use of a simple linear readout has connections to the biological concept of mixed selectivity, as demonstrated in monkeys [8]. The attractiveness of the RC scheme is that naturally occurring dynamical systems (with some desired properties) in physics and engineering can be exploited for temporal information processing without precise tuning of its parameters, circumventing the expensive training cost in alternative schemes, such as recurrent neural networks with tunable internal weights [9]. The ease of RC implementation has brought forward many successful hardware implementations of classical (i.e., not quantum mechanical) RC schemes [10–12]. A spintronic RC achieved state-of-the-art performance on a spoken digit recognition task [13] and a photonic RC demonstrated high-speed speech classification with a low error [14]. For theoretical developments, Gonon *et al.* [15] derived an approximation error

*h.nurdin@unsw.edu.au

upper bound for certain classical RCs on learning a class of input-output maps (not necessarily fading memory maps considered here). Information processing capacity of various RC schemes has been investigated [16,17]. See Refs. [18,19] for further interests and developments of RCs.

In this work, we employ dissipative quantum systems as quantum reservoirs (QRs) to approximate nonlinear input-output maps with fading memory. A map has fading memory if its outputs depend increasingly less on inputs from earlier times. These maps are important in a broad class of real-world problems, including spoken digit recognition [13] and neural modeling [20]. The use of quantum systems as QRs was initially proposed in Refs. [21,22] to harness disordered-ensemble quantum dynamics for temporal information processing. This QR class is suitable for ensemble quantum systems and a static (nontemporal) version of Ref. [21] was demonstrated in NMR to approximate static maps [23]. However, it remained an open problem to show this QR class has the properties required for reservoir computing. Chen and Nurdin [24] addressed this problem by demonstrating that a variation of the scheme proposed in Refs. [21,22] is universal for nonlinear fading memory maps, meaning that given any target nonlinear fading memory map, there exists a member in the universal QR class whose outputs approximate the target map's outputs arbitrarily closely and uniformly over the input sequences. This is a quantum analogue of the universal function approximation property feedforward neural networks enjoy [25,26], but for nonlinear fading memory mappings from input sequences to output sequences. The notion of universality we adopt here was previously established for classical RC schemes [20,27,28] and the Volterra series [29]. In particular, Grigoryeva and Ortega [28] proved this universality property for a form of recurrent neural networks called echo-state networks [30]. However, realizing these previous QR proposals in the quantum gate model remains challenging due to the large number of quantum gates required to implement the dynamics via Trotterization.

The contribution of this work is twofold. Firstly, we propose a class of QRs endowed with the fading memory and universality properties that is not implemented by Ising Hamiltonians, circumventing the need for Trotterization required in previous proposals. Secondly, we propose a realization of a subclass of the universal QR class on NISQ devices and present proof-of-principle experiments on remotely accessed IBM superconducting quantum processors [31], i.e., NISQ devices not yet equipped with quantum error correction. The QR dynamics in this subclass can be implemented using arbitrary but fixed quantum circuits, as long as they generate nontrivial dynamics. This could be, for instance, quantum circuits that are classically intractable to simulate. The quantum circuits can be of short lengths and can be implemented using parameterized single-qubit and multi-qubit quantum gates native to

the quantum hardware, without the need for precise tuning of their gate parameters. Our proof-of-principle experiments show that QRs with a small number of qubits operating in a noisy environment can tackle complex nonlinear temporal tasks, even under current hardware limitations and in the absence of readout and process error mitigation techniques. This work serves as a theoretical and experimental realization of applying near-term gate-model quantum computers to nonlinear temporal information processing tasks, opening an avenue for time series modeling and signal processing applications of these devices.

The rest of this paper is organized as follows. In Sec. II we introduce fading memory maps and describe two temporal information processing tasks for these maps. In Sec. III we introduce the RC framework and explain conditions for which a RC defines a fading memory map. In Sec. IV we present our QR proposal and the universality result. We then propose a subclass of the universal class suitable for implementation on current noisy gate-model quantum computers. We conclude the section by discussing invariance properties of the universal class under certain hardware imperfections. In Sec. V we detail two hardware realizations of the aforementioned subclass of the universal one and present more efficient versions of both schemes that could enable the QR's potential for more scalable temporal processing on gate-model quantum devices. In Sec. VI we detail our proof-of-principle experiments performed on cloud-based IBM superconducting quantum devices. We provide concluding remarks in Sec. VII. Detailed mathematical derivations and experimental settings are provided in the Appendices.

II. TEMPORAL INFORMATION PROCESSING

We consider an input-output (I/O) map M that maps infinite input sequences $u = \{\ldots, u_{-1}, u_0, u_1, \ldots\}$ to infinite output sequences $y = \{\ldots, y_{-1}, y_0, y_1, \ldots\}$, where u_l, $y_l \in \mathbb{R}$ for $l \in \mathbb{Z}$ and $y_l = M(u)_l$ is the output at time l. We write $u|_{L:L'} = \{u_L, \ldots, u_{L'}\}$ and $y|_{L:L'} = \{y_L, \ldots, y_{L'}\}$ to denote the inputs and outputs during time $l = L, \ldots, L'$. In practice, such I/O maps can be realized by convergent dynamical systems, that is, systems that forget their initial condition (see Appendix A 1 for details). If such a dynamical system with state $\mathbf{x}_l$ is initialized at time l_0 at the state $\mathbf{x}_{l_0}$ and given an input sequence $\{u_{l_0}, u_{l_0+1}, \ldots\}$ and the system outputs the sequence $\{y_{l_0}, y_{l_0+1}, \ldots\}$, then it realizes an I/O map M for any initial condition $\mathbf{x}_{l_0}$ as $l_0 \to -\infty$.

Two challenging temporal information processing problems are posed to learn the I/O relationship given by M based on the I/O pair u, y. The first is the multi-step ahead prediction problem, in which we are given inputs $u|_{1:L}$ and the corresponding outputs $y|_{1:L}$. The first $L_T < L$ input-output data pair $(u|_{1:L_T}, y|_{1:L_T})$ is the train

data. In the sequel, we use the input-output train data during $l = 5, \ldots, L_T$. The reason for this is to remove the transient response in the data; see Sec. VI B for a discussion. The goal is to use the train data to optimize the parameters $\mathbf{w}$ of another I/O map $\overline{M}_{\mathbf{w}}$, so that the outputs $\overline{y}|_{L_T+1:L} = \{\overline{y}_{L_T+1}, \ldots, \overline{y}_L\}$, where $\overline{y}_l = \overline{M}_{\mathbf{w}}(u)_l$, approximate the target outputs $y|_{L_T+1:L}$. The second problem is the map emulation problem, that is, to optimize $\mathbf{w}$ of $\overline{M}_{\mathbf{w}}$ to emulate M using $k = 1, \ldots, K$ different I/O train data pairs $(u^k|_{1:L'}, y^k|_{1:L'})$, so that the total number of train data is KL' (we will again use train data during $l = 5, \ldots, l = L'$ in Sec. VI B). When given a previously unseen input $u^{K+1}|_{1:L'}$, the task is for $\overline{y}^{K+1}|_{1:L'}$ to approximate $y^{K+1}|_{1:L'}$.

If an I/O map M has fading memory then its output at time l' becomes increasingly less dependent on input samples u_l from much earlier times $l \ll l'$; see Appendix A 2. In this work, we approximate nonlinear fading memory I/O maps using RCs implemented by quantum dynamical systems. We introduce conditions for which a reservoir dynamical system defines a fading memory I/O map in the next section.

III. RESERVOIR COMPUTING

To approximate fading memory maps, a RC exploits nonlinear dynamical systems to project the input u_l into a reservoir state $\mathbf{x}_l$ at time l. A RC is governed by dynamics f with state evolution $\mathbf{x}_l = f(\mathbf{x}_{l-1}, u_l)$. The dynamics of the reservoir can be arbitrary but fixed as long as it satisfies some required properties, and never requires training. We require the RC to satisfy the echo-state property [30] or the convergence property [32], so that the RC asymptotically forgets its initial condition. The tunable parameters $\mathbf{w}$ appear in a readout function $h_{\mathbf{w}}$, which combines the elements of $\mathbf{x}_l$ into an output $\overline{y}_l = h_{\mathbf{w}}(\mathbf{x}_l)$. For a sufficiently long input sequence $\{u_{l_0}, u_{l_0+1}, \ldots, u_0\}$, the effect of the RC's initial condition can be washed out. As discussed in Sec. II, as $l_0 \to -\infty$, the combination of convergent RC dynamics f and the readout function $h_{\mathbf{w}}$ produces an I/O map $\overline{M}_{(f, h_{\mathbf{w}})}$. After the washout, the readout parameters $\mathbf{w}$ can be optimized using linear regression to minimize an empirical mean squared error between $y_{1:L_T}$ and $\overline{y}_{1:L_T}$. As in previous works [20,27–29], we consider that $\overline{M}_{(f, h_{\mathbf{w}})}$ has the fading memory property.

Echo-state networks, one of the pioneering classical RC schemes, have been numerically demonstrated to achieve state-of-the-art performance in chaotic system modeling [30]. Subsequent hardware realizations of RC proposals exploit classical dynamical systems for real-time temporal processing tasks that demand less energy or computational memory [10–14]. These experiments also suggest empirically that, for certain tasks, such as spoken digit recognition, the reservoir state dimension plays a role in the RC's task performance.

IV. UNIVERSAL QUANTUM RESERVOIR COMPUTERS

We propose to use a QR, with a view towards possibly taking advantage of fast quantum dynamics and its exponentially large state space. A QR consists of N noninteracting subsystems, each subsystem k has an n_k number of qubits so that the QR has $n = \sum_{k=1}^{N} n_k$ qubits. The QR density operator ρ_l at time l evolves according to

$$\rho_l = T(u_l)\rho_{l-1} = \bigotimes_{k=1}^{N} T^{(k)}(u_l)\rho_{l-1}^{(k)}, \quad (1)$$

and the kth subsystem density operator $\rho_l^{(k)}$ undergoes the evolution

$$
\begin{aligned}
&T^{(k)}(u_l)\rho_{l-1}^{(k)} \\
&= (1 - \epsilon_k)[u_l T_0^{(k)} + (1 - u_l)T_1^{(k)}]\rho_{l-1}^{(k)} + \epsilon_k \sigma_k \quad (2)
\end{aligned}
$$

for input $0 \leq u_l \leq 1$. Here, $0 < \epsilon_k \leq 1$, σ_k is an arbitrary but fixed density operator, and $T_0^{(k)}$ and $T_1^{(k)}$ are two arbitrary but fixed completely positive trace-preserving (CPTP) maps. Examples of such maps include some naturally occurring noisy quantum channels, such as dephasing or amplitude damping channels; see Ref. [33]. No precise tuning or engineering of the CPTP maps $T_0^{(k)}, T_1^{(k)}$ is required for the QR scheme and it should not generate trivial dynamics (i.e., we should not choose $T_0^{(k)} = T_1^{(k)}$). They could potentially be classically intractable to simulate CPTP maps. The QR dynamics given by Eqs. (1)–(2) are convergent, meaning that they will asymptotically forget their initial condition; see Appendix A 1 for the proof. Given inputs $\{u_{l_0}, u_{l_0+1}, \ldots, u_0\}$ and $l_0 \to -\infty$, the convergence property ensures that the QR state ρ_0 that evolves according to Eqs. (1)–(2) is determined by $\{u_{l_0}, u_{l_0+1}, \ldots, u_0\}$ and $T_0^{(k)}, T_1^{(k)}$, but not by its initial state ρ_{l_0}.

We obtain partial information about ρ_l by measuring each qubit in the Pauli Z basis to obtain $\langle Z^{(i)} \rangle_l = \text{Tr}(\rho_l Z^{(i)})$ for $i = 1, \ldots, n$, where $Z^{(i)}$ acts on qubit i. We associate the readout function (3) to the QR dynamics (1). The readout function (3) is a multivariate polynomial of degree R in the variables $\langle Z^{(i_j)} \rangle_l$. A simple linear form ($R = 1$) is employed in our proof-of-principle experiments in Sec. VI. The tunable readout parameters $\mathbf{w} = \{\mathbf{w}_{i_1, \ldots, i_n}^{r_{i_1}, \ldots, r_{i_n}}, \mathbf{w}_c\}$ can be optimized via linear regression. Equations (1) and (3) define a QR implementing an I/O map $\overline{M}_{(T, h_{\mathbf{w}})}$ that depends on the QR dynamics T and the readout function $h_{\mathbf{w}}$. We show in Appendix A 2 that $\overline{M}_{(T, h_{\mathbf{w}})}$ has the fading memory property. Now consider the class $\mathcal{M}$ of I/O maps $\overline{M}_{(T, h_{\mathbf{w}})}$ arising from differing numbers of subsystems N, numbers of qubits n, QR dynamics $T(u_l)$, readout parameters $\mathbf{w}$, and degree R of $h_{\mathbf{w}}$. Our main result shows that the class $\mathcal{M}$

is universal for approximating nonlinear fading memory maps.

Theorem 1 (Universality). *Let $K([0, 1])$ be the set of input sequences $\{u_l\}$ with $0 \le u_l \le 1$ for $l \in \mathbb{Z}$. For any nonlinear fading memory map M and any $\delta > 0$, there exists $\overline{M}_{(T,h_\mathbf{w})} \in \mathcal{M}$ implemented by some QR such that, for all $u \in K([0, 1])$, $\sup_{l \in \mathbb{Z}} |M(u)_l - \overline{M}_{(T,h_\mathbf{w})}(u)_l| < \delta$.*

We remark that universality is a property of the QR class $\mathcal{M}$ and not of an individual member of $\mathcal{M}$. The universality proof employs the Stone-Weierstrass theorem [[34], Theorem 7.3.1]; see Appendix A 3 for the proof. Besides the universality property, our proposed universal QR class exhibits invariance properties under certain hardware imperfections; see Sec. IV B.

The readout function associated with the QR dynamics (1) and (2) is

$$
\begin{aligned}
\overline{y}_l &= h_\mathbf{w}(\rho_l) \\
&= \sum_{d=1}^{R} \sum_{i_1=1}^{n} \cdots \sum_{i_n=i_{n-1}+1}^{n} \sum_{r_{i_1}+\cdots+r_{i_n}=d} \mathbf{w}_{i_1,\ldots,i_n}^{r_{i_1},\ldots,r_{i_n}} \langle Z^{(i_1)} \rangle_l^{r_{i_1}} \\
&\quad \cdots \langle Z^{(i_n)} \rangle_l^{r_{i_n}} + \mathbf{w}_c.
\end{aligned} \tag{3}
$$

A. A subclass implementable on noisy gate-model quantum devices

With a limited number of qubits and other current quantum hardware restrictions, not all QR dynamics of the form (1)–(2) can be efficiently implemented. Here we describe a subclass of the universal QR class implementable on current gate-model quantum devices.

QRs in this subclass are governed by Eqs. (1)–(2) with unitary evolutions $T_j^{(k)}(\rho_{l-1}^{(k)}) = U_j^{(k)} \rho_{l-1}^{(k)} U_j^{(k)\dagger}$, $j = 0, 1$, where the unitaries $U_0^{(k)}$ and $U_1^{(k)}$ are arbitrary but fixed. In practice, $U_j^{(k)}$ can be implemented by native quantum gates of NISQ devices, possibly composed of single-qubit and multi-qubit gates, each parameterized by some gate parameter. These gate parameters can be chosen arbitrarily but fixed and should not generate trivial dynamics (e.g., we should not have $U_0^{(k)} = U_1^{(k)}$); thus, precise tuning of these parameters is not required. In Sec. VI A, we suggest some natural choices of $U_j^{(k)}$ tailored for the cloud-based IBM quantum devices [31]. The QR dynamics in this subclass have a natural quantum circuit interpretation; see Fig. 1. The state $\rho(u_l)$ encodes the input u_l as a classical mixture $\rho(u_l) = u_l|0\rangle\langle0| + (1 - u_l)|1\rangle\langle1|$, meaning that we apply $U_0^{(k)} \rho_{l-1}^{(k)} U_0^{(k)\dagger}$ with probability u_l, and apply $U_0^{(k)\dagger} U_0^{(k)} U_1^{(k)} \rho_{l-1}^{(k)} U_1^{(k)\dagger} U_0^{(k)\dagger} U_0^{(k)} = U_1^{(k)} \rho_{l-1}^{(k)} U_1^{(k)\dagger}$ with probability $1 - u_l$. Let $\overline{\rho}_{l-1}^{(k)}$ denote the QR's kth subsystem state after these operations. The state ρ_{ϵ_k} is a classical mixture $\rho_{\epsilon_k} = (1 - \epsilon_k)|0\rangle\langle0| + \epsilon_k|1\rangle\langle1|$ that encodes
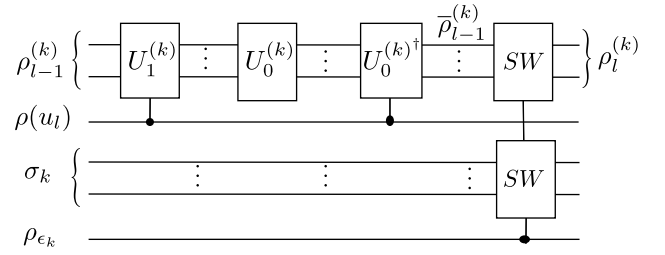

FIG. 1. Quantum circuit interpretation of the QR universal subclass described in Sec. IV A. Here $\rho_{l-1}^{(k)}$ and σ_k are two quantum registers (i.e., groups of qubits), whereas $\rho(u_l)$ and ρ_{ϵ_k} are two single-qubit states. The unitaries $U_1^{(k)}$ and $U_0^{(k)\dagger}$ act on $\rho_{l-1}^{(k)}$, controlled by $\rho(u_l)$. The right-most operation (the SW) swaps the states of $\overline{\rho}_{l-1}^{(k)}$ and σ_k, controlled by ρ_{ϵ_k}.

the rate ϵ_k at which the kth subsystem forgets its initial conditions. That is, with probability ϵ_k, the states $\overline{\rho}_{l-1}^{(k)}$ and σ_k are exchanged, equivalent to resetting the state $\overline{\rho}_{l-1}^{(k)}$ to the fixed density operator σ_k; otherwise, the state $\overline{\rho}_{l-1}^{(k)}$ is unchanged with probability $1 - \epsilon_k$. We again associate the readout function (3) to this QR subclass.

B. Invariance under stationary Markovian hardware noise and time-invariant readout error

The QR dynamics (1) are invariant under stationary Markovian noise. A stationary Markovian noise process acting on the kth subsystem during some time interval $\tau(l - 1) \le t \le \tau l$, where l is the time step and $\tau > 0$, can be modeled as a CPTP map $\mathcal{T}^{(k)}$ for all $l \ge 0$. The kth subsystem's dynamics (2) under this noise process is

$$
\begin{aligned}
\rho_l^{(k)} = (1 - \epsilon_k)[u_l \mathcal{T}^{(k)} \circ T_0^{(k)} + (1 - u_l)\mathcal{T}^{(k)} \circ T_1^{(k)}]\rho_{l-1}^{(k)} \\
+ \epsilon_k \mathcal{T}^{(k)}(\sigma_k),
\end{aligned}
$$

where $\mathcal{T}^{(k)} \circ T_j^{(k)}$ is again some CPTP map for $j = 0, 1$ and $\mathcal{T}^{(k)}(\sigma_k) = \sigma_k'$ is again some fixed density operator. The resulting noisy dynamics again have the form (2) and the form of QR dynamics (1) also remains unchanged. That is, the universal family $\mathcal{M}$ is *invariant and remains universal* under stationary Markovian noise. For hardware implementation of the QR subclass described in Sec. IV A, if the hardware noise is stationary and Markovian then it acts to replace $U_j^{(k)} \rho_{l-1}^{(k)} U_j^{(k)\dagger}$ with another CPTP map $\overline{T}_j^{(k)}(\rho_{l-1}^{(k)})$. The resulting noisy QR dynamics are again of the form (1).

The stationary Markovian noise model is the noise model adopted in the IBM Qiskit simulator [35,36]. The Qiskit noisy simulation approximates the hardware noise as a CPTP map being applied after the application of a unitary gate. The noise parameters are estimated during periodic calibrations on the hardware. Between two calibrations, the calibrated noise parameters remain unchanged and the noisy simulation approximates the hardware noise

by a stationary Markovian noise model. However, during the experiments, the underlying hardware noise could potentially be time varying. Considering these factors, the agreement between our experimental and the Qiskit noisy simulation results (see Appendix E 4 for the data) indicate that the underlying hardware noise approximately preserves the QR dynamics of the form (1) during the experiments. If the underlying noise is nonstationary but changes slowly, the QR output weights can be retrained periodically using the most recently gathered data. This remains a challenge to demonstrate on current cloud access only NISQ devices, but may be possible on future NISQ machines.

Furthermore, QR predicted outputs remain unchanged under time-invariant readout error whenever a linear readout function is used [i.e., $R = 1$ in Eq. (3), which is often employed in practice and in our proof-of-principle experiments]. This is because time-invariant readout error introduces a time-invariant linear transformation of the measurement data and, if the output weights $\mathbf{w}_{i_1,\ldots,i_n}^{r_{i_1},\ldots,r_{i_n}}$ and $\mathbf{w}_c$ are optimized via linear regression, the resulting QR predicted outputs $\bar{y}_l$ remain unchanged; see Appendix B for the derivation.

V. REALIZATION OF A SUBCLASS ON CURRENT QUANTUM HARDWARE

We present two implementation schemes of the subclass described in Sec. IV A on current gate-model quantum computers, such as on the IBM superconducting quantum devices. The first scheme takes into account limitations of some current hardware, and the second scheme employs quantum nondemolition (QND) measurements to substantially reduce the number of circuit runs required. We further show that the QR's convergence property leads to more efficient versions of both schemes. Here, we focus on n-qubit QRs with a single subsystem [$N = 1$ in Eq. (1)] and drop the subsystem index k in Eq. (2). The case with multiple subsystems ($N > 1$) is a straightforward extension. We may choose $\sigma = |\psi\rangle\langle\psi|$ with an easy to prepare pure state $|\psi\rangle$. In all schemes, we initialize the QR circuits in $|0\rangle^{\otimes n}$.

The first implementation follows from an earlier work [[37], Sec. III] and is employed in our proof-of-principle experiments (see Sec. VI). We consider NISQ devices that allow pure state preparation. Instead of realizing Fig. 1 that requires mixed-state preparation, we efficiently implement QRs through Monte Carlo sampling. We construct N_m circuits such that, for each circuit and at each timestep l, we apply U_0 and U_1 with probabilities $(1 - \epsilon)u_l$ and $(1 - \epsilon)(1 - u_l)$, respectively; otherwise, the circuit is set in $|\psi\rangle$ with probability ϵ. Therefore, for each N_m circuit and each time l, implementing the input-dependent QR dynamics $T(u_l)$ in Eq. (1) amounts to applying the gate sequence realizing U_0 or U_1, or resetting the circuit in $|\psi\rangle$. As

N_m is increased, the average of all measurements gives a more accurate estimate of the true expectation $\langle Z^{(i)}\rangle_l$. Furthermore, some current NISQ devices do not allow qubit reset, meaning that once a qubit is measured, it cannot be reused for computation. To estimate $\langle Z^{(i)}\rangle_l$, we reinitialize N_m circuits in $|0\rangle^{\otimes n}$ and reapply $T(u_k)$ from time $k = 1$ to time $k = l$, and only measure $Z^{(i)}$ at the final time l. Each of the N_m circuits is run for S shots at each time l. To process a length-L input sequence under the pure state and qubit reset limitations requires $N_m S L$ circuit runs and $N_m S(1 + \cdots + L) = N_m S(L + 1)L/2$ applications of $T(u_l)$.

If qubit reset is available, a more efficient scheme using QND measurements [38] can be realized; see Appendix C for the details. We no longer need to rerun the N_m circuits from time 1 to estimate $\langle Z^{(i)}\rangle_l$. Instead we just run each of the N_m circuits S shots, meaning that, for each circuit, we perform a QND measurement of $Z^{(i)}$ at time l, continue running the circuit until the next measurement, and so forth. QND measurements ensure information encoded in ρ_l is retained from one timestep to the next. This scheme requires $N_m S L$ applications of $T(u_l)$, but only $N_m S$ circuit runs as opposed to $N_m S L$ runs in the first scheme. We remark that a recent noisy quantum device is equipped with the qubit reset functionality [39], and it will be interesting to implement this scheme in such a device in a future work.

The QR's convergence property (see Appendix A 1) leads to more efficient versions of both schemes. Let $M \geq 1$ be a fixed integer and suppose that we want to estimate $\langle Z^{(i)}\rangle_l$ at a sufficiently large time l (that depends on ϵ, i.e., the rate of forgetting the initial condition). Suppose that we initialize N_m circuits in $|0\rangle^{\otimes n}$, and reapply and rerun $T(u_k)$ from $k = 1$ as before. We then obtain the QR states ρ_{l-M} at time $l - M$ and ρ_l at time l. Thanks to the convergence property, we can instead reinitialize the N_m circuits in $|0\rangle^{\otimes n}$ at time $l - M$ and, from this time onwards, reapply and rerun $T(u_k)$ according to inputs $\{u_{l-M+1}, \ldots, u_l\}$. At time l, we have the corresponding QR state $\tilde{\rho}_l$. By the convergence property (see Appendix C for the derivation), we can make the difference between ρ_l and $\tilde{\rho}_l$ negligible by choosing M appropriately based on ϵ. If we perform repeated measurements on ρ_l and $\tilde{\rho}_l$, the estimates of $\langle Z^{(i)}\rangle_l$ and $\langle \widetilde{Z^{(i)}}\rangle_l = \text{Tr}(\tilde{\rho}_l Z^{(i)})$ will also be close; see Appendix D.

The convergence property can be readily exploited on current NISQ machines, leading to efficient versions of both schemes. The first scheme now requires $N_m S L$ circuit runs but only $N_m S M$ applications of $T(u_l)$. The second scheme now only needs $N_m S$ circuit runs and $N_m S M$ applications of $T(u_l)$, both are *independent* of the input length L, enabling the QR's potential for fast and scalable temporal processing. In all schemes, it is possible and perhaps advantageous to set $S = 1$ and run N_m circuits (possibly in parallel if multiple copies of the same hardware are available) for a sufficiently large N_m. The average of N_m measurements estimates $\langle Z^{(i)}\rangle$, whose estimation accuracy

increases as N_m increases; see Appendix D for the analysis. Since qubit reset is not yet available on the IBM superconducting quantum devices, we employ the first implementation scheme in our proof-of-principle experiments. It will be a future work of interest to realize these more efficient protocols on gate-model quantum hardware.

VI. PROOF-OF-PRINCIPLE EXPERIMENTS

Five nonlinear tasks are chosen to carefully test different computational aspects of the QR proposal. Tasks I–IV have the fading memory property. Tasks I and II test the QR's ability to learn high-dimensional nonlinear maps. Both tasks are governed by linear dynamics determined by some matrix A and have the same form of nonlinear output. The maximum singular value $\sigma_{\max}(A)$ determines the rate at which the dynamics forget their initial condition, while the sparsity of A reflects the pairwise correlation of the reservoir state elements. Task I is described by a dense matrix A with $\sigma_{\max}(A) = 0.5$ and task II is governed by A with 95% sparsity with $\sigma_{\max}(A) = 0.99$. Task III tests the QR's ability to learn nonlinear maps governed by highly nonlinear dynamics. Task IV tests the short-term memory ability and task V is a long-term memory map for testing the capability of the QR beyond its theoretical guarantee. For all experimental and numerical details, see Appendix E.

We implement four distinct QRs from the subclass described in Sec. IV A on three IBM superconducting quantum processors [31]. Each QR consists of a single subsystem [$N = 1$ in Eq. (1)] with a linear output function [$R = 1$ in Eq. (3)]. Hereafter, we drop the subsystem index k. The 4-qubit and 10-qubit QRs are implemented on the 20-qubit Boeblingen device; qubits with lower gate errors and longer coherence times are chosen. The 5-qubit Ourense and Vigo devices are used for two distinct 5-qubit QRs. These 5-qubit quantum devices admit simpler qubit couplings but lower gate errors than the 20-qubit Boeblingen device; see Appendix E 5 for hardware specifications. Through comparison among the four QRs, we can investigate the impact of the size of QRs, the complexity of quantum circuits implementing the QR dynamics, and the intrinsic hardware noise on the QRs' approximation performance.

A. Quantum circuits for QRs

We require the QRs to forget initial conditions for approximating fading memory maps. Traditionally, initial conditions are washed out with a sufficiently long input sequence until reaching a steady state. Here we bypass the washout by choosing $\sigma = (|0\rangle\langle0|)^{\otimes n}$ and U_0 so that $|0\rangle^{\otimes n}$ is the steady state of Eq. (1) under $u_l = 1$, meaning that we can initialize the QR circuits in $|0\rangle^{\otimes n}$. Furthermore, U_0 and U_1 should be different and hardware efficient but sufficiently complex to produce nontrivial quantum dynamics.
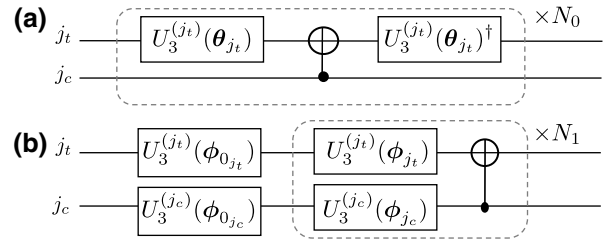

FIG. 2. Quantum circuit schematics for (a) $U_0(\boldsymbol{\theta})$ and (b) $U_1(\boldsymbol{\phi})$ employed in proof-of-principle experiments, described by Eq. (4) in Sec. VI A. Here j_t and j_c are the target and control qubits, respectively. The unitaries $U_0(\boldsymbol{\theta}), U_1(\boldsymbol{\phi})$ consist of N_0, N_1 layers of highlighted gate operations, with each layer acting on a different qubit pair (j_t, j_c).

We choose the circuit schematics [see also Figs. 2(a) and 2(b)]

$$U_0(\boldsymbol{\theta}) = \prod_{j=1}^{N_0}[U_3^{(j_t)}(\boldsymbol{\theta}_{j_t})\mathrm{CX}_{j_cj_t}U_3^{(j_t)}(\boldsymbol{\theta}_{j_t})^\dagger],$$

$$U_1(\boldsymbol{\phi}) = \bigotimes_{i=1}^{n} U_3^{(i)}(\boldsymbol{\phi}_{0_i}) \prod_{j=1}^{N_1}\left[\bigotimes_{i=1}^{n} U_3^{(i)}(\boldsymbol{\phi}_{j_i})\mathrm{CX}_{j_cj_t}\right],$$

(4)

where $\boldsymbol{\theta}_{j_t} = (\boldsymbol{\theta}_{j_t}^0, \boldsymbol{\theta}_{j_t}^1, \boldsymbol{\theta}_{j_t}^2)$ and $\boldsymbol{\phi}_{j_i} = (\boldsymbol{\phi}_{j_i}^0, \boldsymbol{\phi}_{j_i}^1, \boldsymbol{\phi}_{j_i}^2)$ are gate parameters, each independently and uniformly randomly sampled from $[-2\pi, 2\pi]$. Here $U_3^{(i)}$ is an arbitrary rotation on single qubit i [40] with inverse $U_3^{(j_t)}(\boldsymbol{\theta}_{j_t})^\dagger = U_3^{(j_t)}(-\boldsymbol{\theta}_{j_t}^0, -\boldsymbol{\theta}_{j_t}^2, -\boldsymbol{\theta}_{j_t}^1)$, and $\mathrm{CX}_{j_cj_t}$ is the CNOT gate with control qubit j_c and target qubit j_t. These quantum gates are native to the aforementioned IBM superconducting quantum processors, meaning that no further decomposition into simpler gates is required to implement these chosen gates [31]. The numbers of layers N_0 and N_1 are sufficiently large to couple all qubits linearly while respecting the coherence limits of these devices. Owing to the more flexible qubit couplings in the Boeblingen device, circuits implementing the 4-qubit and 10-qubit QRs have more gate and random parameters than the 5-qubit QRs.

For the 4-qubit and 10-qubit QRs on the Boeblingen device, we choose the number of layers $N_0 = N_1 = 5$ in Eq. (4). For the 5-qubit Ourense QR, we implement a simpler form of Eq. (4), given by

$$U_0 = \prod_{j=1}^{4}\mathrm{CX}_{j_cj_t}, \qquad U_1(\boldsymbol{\phi}) = \bigotimes_{i=1}^{5} U_3^{(i)}(\boldsymbol{\phi}_i).$$
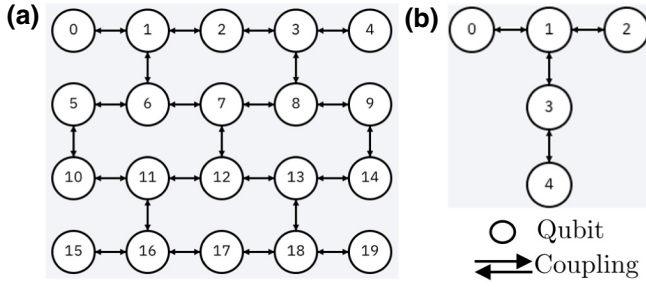
(a)

(b)

FIG. 3. Qubit coupling maps of the IBM superconducting quantum processors. (a) The 20-qubit Boeblingen device. (b) Both the 5-qubit Ourense and Vigo devices.

To implement different QR dynamics on the 5-qubit Vigo device, we choose

$$U_0(\boldsymbol{\theta}) = \prod_{j=1}^{3} [R_y^{(j_t)}(\boldsymbol{\theta}_{j_t}) \mathrm{CX}_{j_c j_t} R_y^{(j_t)}(\boldsymbol{\theta}_{j_t})^{\dagger}],$$

$$U_1(\boldsymbol{\phi}) = \bigotimes_{i=1}^{5} R_x^{(i)}(\boldsymbol{\phi}_i).$$

Here $R_y^{(i)}$ and $R_x^{(i)}$ are rotational Y and X gates on qubit i, respectively. Both gates are special instances of the arbitrary single-qubit rotational gate $U_3^{(i)}$ with one (free) gate parameter and the other two fixed constants. For all QRs, natively coupled control and target qubits for CNOT gates are chosen, meaning that a CNOT gate can be directly applied to the qubit pair without additional gate operations. See Fig. 3 for the device qubit coupling maps and Appendix E 2 for the QR quantum circuit details.

B. Experimental implementation

In this section we report on experiments demonstrating the first implementation scheme described in Sec. V. We choose a sufficiently large $N_m = 1024$ and $\epsilon = 0.1$ for a moderate short-term memory. To estimate $\langle Z^{(i)} \rangle_l$ at time l, each of the N_m circuits implementing the QRs on the Boeblingen device and the 5-qubit QRs are run for $S = 1024$ and $S = 8192$ shots, respectively. These shot numbers are chosen according to circuit execution times of the devices.

We apply the four QRs to the five nonlinear tasks on the multistep ahead prediction and map emulation problems. To implement the same washout as for the QRs for each target map, we inject a constant input sequence $u_l = 1$ of length 50 followed by train and test inputs uniformly randomly sampled from $u_l \in [0, 1]$. This change in the input statistics leads to a transitory target output response. We remove the associated transients by discarding the first four target input-output data and the corresponding QR experimental data; see Appendix E 3 for all data. For the multistep ahead problem, train and test time steps run from

$l = 5$ to $L_T = 23$ and $L_T + 1 = 24$ to $L = 30$, respectively. For the map emulation problem, $K = 2$ train input-output pairs running from $l = 5$ to $L' = 24$ are used, followed by one unseen test input-output pair with the same time steps. The number of train and test data in our proof-of-principle experiments is limited by the length of quantum circuits allowed on the IBM quantum processors. Furthermore, these cloud-based quantum processors are shared among users, making continuous experiments infeasible and durations of experiments lengthy. Yet our work indicates that, despite these current limitations, NISQ devices can demonstrate learning of input-output maps, and supports the QR as a viable intermediate application of NISQ machines on the road to full-fledged quantum devices equipped with quantum error correction.

To harness the flexibility of the QR approach, a multitasking technique is used, in which the four QRs are evolved and the estimates of $\langle Z^{(i)} \rangle_l$ for all time steps are recorded once, whereas the readout parameters $\mathbf{w}$ are optimized independently for each task. That is, fixed QR dynamics, with fixed gate parameter values, are exploited for multiple tasks simultaneously. We evaluate and compare the task performance of QRs using the normalized mean squared error ($\mathcal{E}$) between prediction $\bar{y}|_{L_T+1:L}$ and target $y|_{L_T+1:L}$, computed as

$$\mathcal{E} = \sum_{l=L_T+1}^{L} |y_l - \bar{y}_l|^2 / \Delta_y^2,$$

where $\mu = 1/(L - L_T) \sum_{l=L_T+1}^{L} y_l$ and $\Delta_y^2 = \sum_{l=L_T+1}^{L} (y_l - \mu)^2$. While the success of the experimental demonstration of hybrid quantum-classical algorithms often requires error mitigation techniques to reduce the effect of decoherence [41,42], we remark that our results are obtained without any process or readout error mitigation.

C. QR task performance

As the number of qubits increases, the 10-qubit Boeblingen QR is expected to perform better than other QRs. For the multistep ahead prediction problem, we observe that two qubits in the 10-qubit Boeblingen QR experienced significant time-varying deviations between the experimental data and simulation results on the Qiskit simulator; see Appendix E 4 for a discussion. To remedy this issue, we set the corresponding elements of $\mathbf{w}$ to be 0s. The resulting 10-qubit Boeblingen QR (with $\mathcal{E} < 0.08$) outperforms other QRs with a smaller number of qubits on the first four tasks, and achieves an almost two-fold performance improvement on tasks II and III; see Table I for all $\mathcal{E}$s on the multistep ahead prediction problem. The 10-qubit Boeblingen QR predicted outputs follow the target outputs relatively closely, as shown in Fig. 4(a). The 5-qubit

TABLE I. $\mathcal{E}$s on the multistep ahead prediction.

Task	10-qubit Boeblingen	4-qubit Boeblingen	5-qubit Ourense	5-qubit Vigo
I	0.051	0.088	0.24	0.070
II	0.072	0.12	0.68	0.22
III	0.043	0.10	0.25	0.081
IV	0.079	0.092	0.34	0.11
V	0.47	0.41	2.3	0.20

FIG. 5.   The spatial multiplexing schematic. The same input sequence is injected into two distinct 5-qubit QRs. The internal states $\mathrm{Tr}(\rho_l Z^{(i)})$ of the two QRs are linearly combined to form a single output.

Ourense QR admits very simple dynamics, whereas the 5-qubit Vigo QR has more gate operations and gate parameters. The 5-qubit Ourense QR is outperformed by the 5-qubit Vigo QR in all tasks. Considering that the Ourense and Vigo devices have similar noise characteristics and the same qubit coupling map, this suggests that the QR performance can be improved by choosing a more complex quantum circuit, in the sense of having a longer gate sequence.

The 10-qubit Boeblingen QR performs better on all tasks than the 5-qubit QRs except on task V. This could be due to the impact of the higher noise level in the Boeblingen device and the fact that the output sequence is generated by a map that is not known to be fading memory; see Appendix E 5 for the hardware specifications. Our universal class of QRs can exploit the property of spatial multiplexing as initially proposed in Ref. [22]; also see Ref. [24] and Fig. 5 for an illustration. Outputs of distinct and noninteracting 5-qubit QRs can be combined linearly to harness the computational features of both members. Since the combined Ourense and Vigo devices have ten qubits overall as with the 10-qubit Boeblingen QR but with
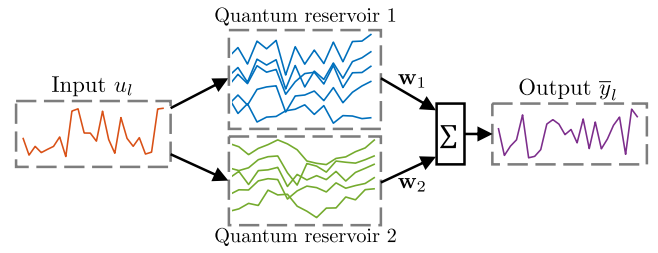
lower noise levels, it would be meaningful to combine the 5-qubit Vigo and Ourense QRs via spatial multiplexing on the map emulation problem. The results of this multiplexing is summarized in Table II.

The combination of two 5-qubit QRs as discussed above achieves $\mathcal{E} = 0.20, 0.13, 0.16, 0.25, 0.20$ for the five tasks without any readout or process error mitigation. The predicted multiplexed QR outputs corresponding to the unseen inputs follow the target outputs relatively closely, as shown in Fig. 4(b). Without spatial multiplexing, the 5-qubit Ourense or the 5-qubit Vigo QR shows a worse performance in the first four tasks; see Table II. The spatial multiplexed 5-qubit QR combines computational features from the constituent QRs and can achieve comparable performance to the individual members as well as gaining an almost two-fold performance boost on tasks II and III. We
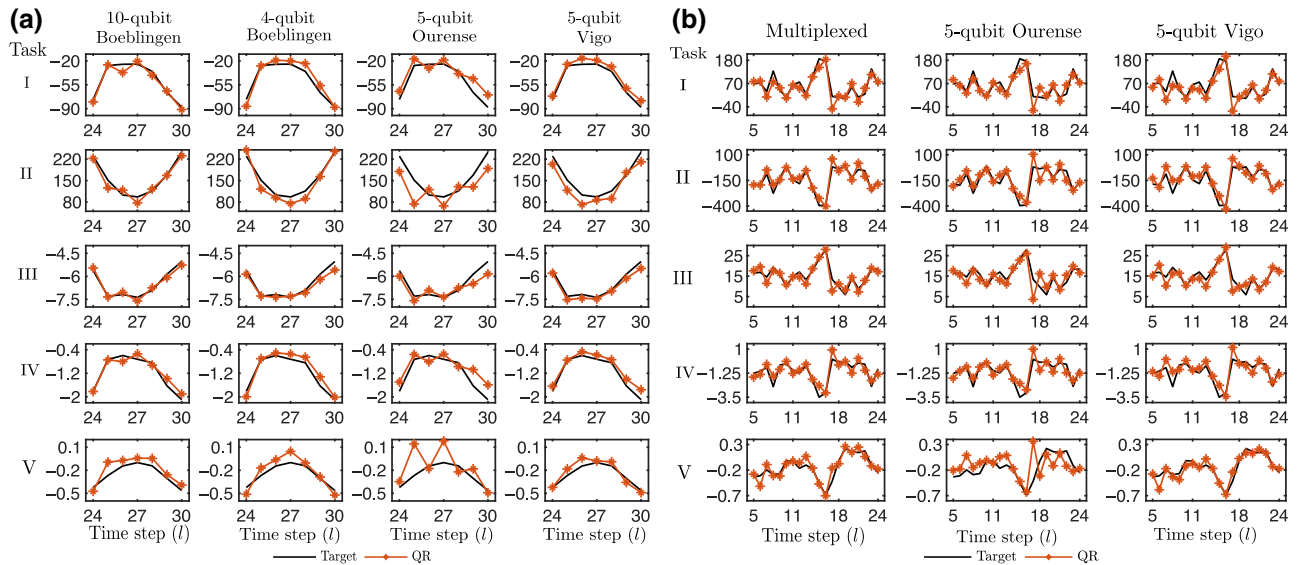

FIG. 4.   The QR's predicted outputs for (a) the multistep prediction problem and (b) the map emulation problem. Rows and columns in (a) correspond to different tasks and QRs, respectively. The first column in (b) corresponds to the multiplexed QR.

TABLE II. $\mathcal{E}$s on the map emulation.

Task	Multiplexed QR	5-qubit Ourense	5-qubit Vigo
I	0.20	0.26	0.32
II	0.13	0.27	0.23
III	0.16	0.46	0.26
IV	0.25	0.30	0.36
V	0.20	1.1	0.17

anticipate that spatial multiplexing of QRs with more complex circuit structures and a larger number of qubits can lead to further performance improvements.

VII. CONCLUSION

We propose a class of quantum reservoir computers endowed with the universality property that is implementable on available noisy gate-model quantum hardware for temporal information processing. Our approach can harness arbitrary but fixed quantum circuits native to noisy quantum processors, without precise tuning of the circuit parameters. Our theoretical analysis is supported by proof-of-concept experiments on current superconducting quantum devices, demonstrating that small-scale noisy quantum reservoirs can perform nontrivial nonlinear temporal processing tasks under current hardware limitations, in the absence of readout and process error mitigation techniques. We also detail more efficient implementation schemes of our QR proposal that could enable the QR's potential for fast and scalable temporal processing. It is a future work of interest to realize these more efficient protocols on quantum hardware. Our work indicates that quantum reservoir computing can serve as a viable intermediate application of NISQ devices on the road to full-fledged quantum computers.

Our approach is scalable in the number of qubits by offloading exponentially costly computations to noisy quantum systems and utilizing classical algorithms with a linear (in the number of qubits) computational cost to process sequential data. Moreover, when implemented on NISQ devices, the microsecond timescale for the evolution of the quantum reservoir suggests its potential for real-time fast signal processing tasks. Guided by our theory, we applied the spatial multiplexing technique initially proposed in Ref. [22], and demonstrate experimentally that exploiting distinct computational features of multiple small noisy quantum reservoirs can lead to a computational boost. As NISQ hardware becomes increasingly accessible and the noise level is continually reduced, we anticipate that the quantum reservoir approach will find useful applications in a broad range of scientific disciplines that employ time series modeling and analysis. We are also optimistic for useful applications to be possible even with a noise level above the threshold for continuous quantum error correction.

APPENDIX A: UNIVERSALITY FOR APPROXIMATING NONLINEAR FADING MEMORY MAPS

We first define notation for the rest of this section. Let $K([0,1])$ be the set of infinite sequences $u = \{\ldots, u_{-1}, u_0, u_1, \ldots\}$ such that $u_l \in [0,1]$ for all $l \in \mathbb{Z}$. Let $K^+([0,1])$ and $K^-([0,1])$ be subsets of $K([0,1])$ for which the indices are restricted to $\mathbb{Z}^+ = \{1, 2, \ldots\}$ and $\mathbb{Z}^- = \{\ldots, -2, -1, 0\}$, respectively. For any complex matrix A, $\|A\|_p = \text{Tr}(\sqrt{A^\dagger A}^p)^{1/p}$ is the Schatten p-norm for some $p \in [1, \infty)$. For any operator T, the induced operator norm is $\|T\|_{p-p} = \sup_{A \in \mathbb{C}^{n \times n}, \|A\|_p = 1} \|T(A)\|_p$. Let $\mathcal{D}(2^n)$ denote the set of $2^n \times 2^n$ density operators.

Consider an input-output map M that maps an infinite input sequence $u \in K([0,1])$ to a real infinite output sequence $y \in K(\mathbb{R})$. We say that M is w-fading memory if there exists a decreasing sequence $w = \{w_0, w_1, \ldots\}$ with $\lim_{l \to \infty} w_l = 0$ such that, for any $u, v \in K^-([0,1])$, we have $|M(u)_0 - M(v)_0| \to 0$ whenever $\sup_{l \in \mathbb{Z}^-} |w_{-l}(u_l - v_l)| \to 0$. Here $M(u)_l = y_l$ is the output sequence at time l. We also require M to be causal and time invariant as in Ref. [24], meaning that the output of M at time l depends only on the input up to and including time l, and its outputs are invariant under time shifts. Now we are interested in approximating M with a time-invariant fading memory map $\overline{M}$ produced by a quantum reservoir computer.

1. The convergence property

Since M is fading memory, the map $\overline{M}$ must also forget its initial condition ρ_0. This is the convergence property [32] or the echo-state property [30]. We give a precise definition here.

Definition 1 (convergence). —*An input-dependent CPTP map T is convergent with respect to input $u \in K([0,1])$ if there exists a sequence $\{\delta_l; l \geq 0\}$ of positive numbers with $\lim_{l \to \infty} \delta_l = 0$ such that, for all $u \in K^+([0,1])$ and any two density operators $\rho_{j,l}$ $(j = 1, 2)$ satisfying $\rho_{j,l} = T(u_l)\rho_{j,l-1}$, it holds that $\|\rho_{1,l} - \rho_{2,l}\|_1 \leq \delta_l$ for $l \geq 0$. If a QR dynamic T is convergent, we call the QR a convergent system.*

Lemma 1. *The QR dynamics given by Eqs. (1) and (2) are convergent with respect to inputs $u \in K([0,1])$.*

First we show that each subsystem governed by Eq. (2) is convergent. For any $\rho, \sigma \in \mathcal{D}(2^n)$, $u_l \in [0, 1]$, and $\epsilon_k \in (0, 1]$, we have

$$
\begin{aligned}
&\|T^{(k)}(u_l)(\rho - \sigma)\|_1 \\
&\quad = (1 - \epsilon_k)\|[u_l T_0^{(k)} + (1 - u_l)T_1^{(k)}](\rho - \sigma)\|_1 \\
&\quad \leq (1 - \epsilon_k)\|\rho - \sigma\|_1 \\
&\quad \leq 2(1 - \epsilon_k),
\end{aligned}
\tag{A1}
$$

where the last inequality follows from Ref. [[33], Theorem 9.2] and the convex combination $u_l T_0^{(k)} + (1 - u_l)T_1^{(k)}$ is again a CPTP map. Now let $\rho_{1,0}$ and $\rho_{2,0}$ be two arbitrary initial density operators; using inequality (A1) L times, we

have

$$
\begin{aligned}
\|\rho_{1,L} - \rho_{2,L}\|_1 &= \left\|\left(\overleftarrow{\prod_{l=1}^{L}} T^{(k)}(u_l)\right)(\rho_{1,0} - \rho_{2,0})\right\|_1 \\
&\leq (1 - \epsilon_k)^L \|\rho_{1,0} - \rho_{2,0}\|_1 \\
&\leq 2(1 - \epsilon_k)^L,
\end{aligned}
$$

where $\overleftarrow{\prod_{l=1}^{L}} T^{(k)}(u_l)$ is the time composition of $T^{(k)}(u_l)$ from right to left.

Second, we show that QR dynamics (1) are convergent by showing that $T(u_l) = \bigotimes_{k=1}^{N} T^{(k)}(u_l)$ is again convergent when the subsystems are initialized in a product state. We apply the same argument as in Ref. [[43], Lemma 5]. Consider two CPTP maps $T^{(1)}(u_l)$ and $T^{(2)}(u_l)$ of the form (2). Let $\rho_{1,0} \otimes \sigma_{1,0}$ and $\rho_{2,0} \otimes \sigma_{2,0}$ be two arbitrary initial product states. Then $T^{(1)}(u_l) \otimes T^{(2)}(u_l)$ is again convergent with respect to all $u \in K([0, 1])$:

$$
\begin{aligned}
\|\rho_{1,L} \otimes \sigma_{1,L} - \rho_{2,L} \otimes \sigma_{2,L}\|_1 &\leq \left\|\left(\overleftarrow{\prod_{l=1}^{L}} T^{(1)}(u_l) \otimes T^{(2)}(u_l)\right)(\rho_{1,0} \otimes \sigma_{1,0} - \rho_{2,0} \otimes \sigma_{1,0})\right\|_1 \\
&\quad + \left\|\left(\overleftarrow{\prod_{l=1}^{L}} T^{(1)}(u_l) \otimes T^{(2)}(u_l)\right)(\rho_{2,0} \otimes \sigma_{1,0} - \rho_{2,0} \otimes \sigma_{2,0})\right\|_1 \\
&= \left\|\left(\overleftarrow{\prod_{l=1}^{L}} T^{(1)}(u_l)\right)(\rho_{1,0} - \rho_{2,0})\right\|_1 \|\sigma_{1,L}\|_1 + \left\|\left(\overleftarrow{\prod_{l=1}^{L}} T^{(2)}(u_l)\right)(\sigma_{1,0} - \sigma_{2,0})\right\|_1 \|\rho_{2,L}\|_1 \\
&\leq 2(1 - \epsilon_1)^L + 2(1 - \epsilon_2)^L.
\end{aligned}
$$

Repeating this argument N times shows that the QR dynamics $T(u_l) = \bigotimes_{k=1}^{N} T^{(k)}(u_l)$ are again convergent.

2. The fading memory property

Associate the readout function (3) to the QR dynamics given by Eqs. (1) and (2). This defines an I/O map $\overline{M}_{(T, h_{\mathbf{w}})}$. This I/O map is causal, meaning that its output $\overline{y}_l$ depends only on $u_{l'}$ for $l' \leq l$. Furthermore, it is time invariant, meaning that $\overline{y}_{\tau + l} = \overline{M}_{(T, h_{\mathbf{w}})}(S_\tau(u))_l$ for all $\tau \in \mathbb{Z}$, where $S_\tau(u) = \{\ldots, u_{\tau-1}, u_\tau, u_{\tau+1}, \ldots\}$ shifts the input sequence by τ. By causality and time invariance, it suffices to consider the outputs $\overline{y}_l$ of $\overline{M}_{(T, h_{\mathbf{w}})}(u)_l$ for $l \leq 0$ and left-infinite inputs $u \in K^-([0, 1])$; see Refs. [24,27,29] for details.

For any $u \in K^-([0, 1])$ and any initial condition $\rho_{-\infty}$,

$$
\overline{M}_{(T, h_{\mathbf{w}})}(u)_0 = h_{\mathbf{w}}\left[\left(\overrightarrow{\prod_{j=0}^{\infty}} T(u_{-j})\right)\rho_{-\infty}\right],
$$

where $\overrightarrow{\prod_{j=0}^{\infty}} T(u_{-j}) = \lim_{N \to \infty} T(u_0) \cdots T(u_{l-N})$ and the limit is pointwise. We can restate the fading memory property in terms of continuity of $\overline{M}_{(T, h_{\mathbf{w}})}$ with respect to a certain norm. Given a null sequence w (i.e., a decreasing

sequence w with $\lim_{l \to \infty} w_l = 0$) and any $u \in K^-([0, 1])$, define a weighted norm $\|u\|_w = \sup_{l \in \mathbb{Z}^-} |u_l| w_{-l}$. The map $\overline{M}_{(T, h_{\mathbf{w}})}$ is w-fading memory if it is continuous in $(K^-([0, 1]), \|\cdot\|_w)$.

Definition 2 (fading memory). —*Given a null sequence w, the set of w-fading memory maps is the set of all continuous functions $C(K^-([0, 1]), \|\cdot\|_w)$ defined on $(K^-([0, 1]), \|\cdot\|_w)$.*

Lemma 2. *For any null sequence w, $\overline{M}_{(T, h_{\mathbf{w}})}$ induced by the QR described by Eqs. (1)–(3) is w-fading memory.*

Using the same argument as in Ref. [[24], Lemma 3], it follows that $\overline{M}_{(T, h_{\mathbf{w}})}$ is w-fading memory if the CPTP map $T^{(k)}(u_l)$ of each subsystem k is continuous with respect to $u_l \in [0, 1]$ for all $k = 1, \ldots, N$. If fact, we show that $T^{(k)}(u_l)$ is uniformly continuous. Let $x, y \in [0, 1]$ and $A \in$

$\mathbb{C}^{2^{n_k} \times 2^{n_k}}$. Then

$$
\begin{aligned}
&\|T^{(k)}(x) - T^{(k)}(y)\|_{1-1} \\
&= \sup_{A \in \mathbb{C}^{2^{n_k} \times 2^{n_k}}, \|A\|_1 = 1} \|[T^{(k)}(x) - T^{(k)}(y)]A\|_1 \\
&= (1 - \epsilon_k)|x - y| \sup_{A \in \mathbb{C}^{2^{n_k} \times 2^{n_k}}, \|A\|_1 = 1} \|T_0^{(k)}(A) - T_1^{(k)}(A)\|_1 \\
&\leq (1 - \epsilon_k)|x - y|(\|T_0^{(k)}\|_{1-1} + \|T_1^{(k)}\|_{1-1}) \\
&\leq 2(1 - \epsilon_k)|x - y|,
\end{aligned}
$$

where the last inequality follows from Theorem 2.1 of Ref. [44]. We remark that Lemma 3 of Ref. [24] is stated with respect to the Schatten $p = 2$ norm, but the same argument holds for the Schatten $p = 1$ norm.

3. The universality property

Now consider the family $\mathcal{M}$ of maps $\overline{M}_{(T,h_\mathbf{w})}$. We state our main universality result.

Theorem 2 (universality). *For any null sequence w, the QR class $\mathcal{M}$ is dense in $C(K^-([0,1]), \|\cdot\|_w)$. That is, given any w-fading memory map $M \in C(K^-([0,1]), \|\cdot\|_w)$ and any $\delta > 0$, there exists $\overline{M}_{(T,h_{\mathbf{w}})} \in \mathcal{M}$ such that, for all $u \in K^-([0,1])$, $\sup_{l \in \mathbb{Z}^-} |M(u)_l - \overline{M}_{(T,h_{\mathbf{w}})}(u)_l| < \delta$.*

We apply the Stone-Weierstrass theorem to show that $\mathcal{M}$ is dense in $C(K^-([0,1]), \|\cdot\|_w)$. It has been shown that the space $(K^-([0,1]), \|\cdot\|_w)$ is a compact metric space [[27], Lemma 2]. We now state the Stone-Weierstrass theorem.

Theorem 3 (Stone-Weierstrass). *Let E be a compact metric space and let C(E) be the set of real-valued continuous functions defined on E. If a subalgebra A of C(E) contains the constant functions and separates points of E, then A is dense in C(E).*

The fact that the family $\mathcal{M}$ forms a polynomial algebra follows from Lemma 5 of Ref. [24] and the observation that, for any QR dynamics $T_1(u_l) = \bigotimes_{k=1}^{N_1} T_1^{(k)}(u_l)$ and $T_2(u_l) = \bigotimes_{k=1}^{N_2} T_2^{(k)}(u_l)$, where each $T_1^{(k)}, T_2^{(k)}$ has the form (2), we again find that $T(u_l)(\rho_1 \otimes \rho_2) = T_1(u_l)\rho_1 \otimes T_2(u_l)\rho_2$ is of the form (1). Furthermore, $T(u_l) = T_1(u_l) \otimes T_2(u_l)$ is again convergent when initialized in a product state of the subsystems. Therefore, the family $\mathcal{M}$ forms a polynomial algebra consisting of w-fading memory maps.

Constant functions can be obtained by setting $\mathbf{w}_{i_1,...,i_n}^{r_{i_1},...,r_{i_n}} = 0$ in Eq. (3). It remains to show that $\mathcal{M}$ separates points in $K^-([0,1])$. That is, for any distinct $u, v \in K^-([0,1])$ with $u_l \neq v_l$ for at least one l, we need to find a map $\overline{M}_{(T,h_\mathbf{w})} \in \mathcal{M}$ such that $\overline{M}_{(T,h_\mathbf{w})}(u)_0 \neq \overline{M}_{(T,h_\mathbf{w})}(v)_0$. We show that we can construct a single-qubit quantum reservoir with this property.

Consider a single-qubit quantum reservoir with a linear readout function ($n = 1, R = 1, N = 1$). For the rest of this proof, we drop the subsystem index. This quantum reservoir consists of one system qubit and one ancilla qubit denoted as ρ_a. Choose the dynamics

$$
\begin{aligned}
\rho_l &= T(u_l)\rho_{l-1} \\
&= (1 - \epsilon)\{u_l \text{Tr}_a[e^{-iH}(\rho_{l-1} \otimes \rho_a^0)e^{iH}] \\
&\quad + (1 - u_l)\text{Tr}_a[e^{-iH}(\rho_{l-1} \otimes \rho_a^1)e^{iH}]\} + \epsilon K_{I/2},
\end{aligned}
\tag{A2}
$$

where $\rho_a^j = |j\rangle\langle j|$ for $j = 0, 1$, Tr_a denotes the partial trace over ancilla ρ_a, and $\epsilon \in (0, 1)$. The map $K_{I/2}$ is a CPTP map defined as $K_{I/2}(X) = \text{Tr}(X)(I/2)$ for any $X \in \mathbb{C}^{2 \times 2}$. The Hamiltonian H is of the Ising type $H = J(X^{(0)}X^{(1)} + Y^{(0)}Y^{(1)}) + \alpha \sum_{j=0}^1 Z^{(j)}$, where $X^{(j)}, Y^{(j)}$, and $Z^{(j)}$ are the Pauli X, Y, and Z operators on qubit j, with $j = 0$ being the ancilla qubit.

The matrix representation of the CPTP map (A2) is

$$
\overline{T}(u_l) = |00\rangle\langle 00| + (1 - \epsilon)\begin{pmatrix} 0 & 0 & 0 & 0 \\ \sin^2(2J)(2u_l - 1) & \cos^2(2J) & 0 & 0 \\ 0 & 0 & \cos(2J)\cos(2\alpha) & -\cos(2J)\sin(2\alpha) \\ 0 & 0 & \cos(2J)\sin(2\alpha) & \cos(2J)\cos(2\alpha) \end{pmatrix}.
\tag{A3}
$$

We order an orthogonal basis for $\mathbb{C}^{2 \times 2}$ as $\{I, Z, X, Y\}$. The matrix representation of the CPTP map (A2) is given by Eq. (A3). Since Eq. (A2) is convergent, we can choose any initial condition $\rho_{-\infty} = |0\rangle\langle 0|$ with the corresponding vector representation $\overline{\rho}_\infty = (1/2)(1, 1, 0, 0)$. Taking a linear readout function, for $u \in K^-([0,1])$, the quantum reservoir implements

$$
\overline{M}_{(T,h_\mathbf{w})}(u)_0 = 2\mathbf{w}_1\left[\left(\overrightarrow{\prod}_{j=0}^\infty \overline{T}(u_{-j})\right)\overline{\rho}_{-\infty}\right]_2 + \mathbf{w}_c,
$$

where $[\cdot]_2$ is the second element of the vector corresponding to $\mathrm{Tr}(Z\rho_0)/2$.

Now given two distinct inputs $u, v \in K^-([0,1])$, suppose that $u_0 \neq v_0$. Then choose J such that $\cos^2(2J) = 0$ and, therefore,

$$\overline{M}_{(T,h_{\mathbf{w}})}(u)_0 - \overline{M}_{(T,h_{\mathbf{w}})}(v)_0 = 2\mathbf{w}_1(1-\epsilon)(u_0 - v_0) \neq 0.$$

Suppose that $u_0 = v_0$, and note that, in general,

$$\overline{M}_{(T,h_{\mathbf{w}})}(u)_0 = \mathbf{w}_1 \sin^2(2J)(1-\epsilon)$$
$$\sum_{j=0}^{\infty}[(1-\epsilon)\cos^2(2J)]^j(2u_{-j}-1).$$

Choose $\epsilon \in (0,1)$ and J such that $(1-\epsilon)\cos^2(2J) \in (0, 1-\epsilon)$. Then the above is a convergent power series and the subtraction is well defined:

$$\overline{M}_{(T,h_{\mathbf{w}})}(u)_0 - \overline{M}_{(T,h_{\mathbf{w}})}(v)_0 = 2\mathbf{w}_1 \sin^2(2J)(1-\epsilon)$$
$$\times \sum_{j=0}^{\infty}[(1-\epsilon)\cos^2(2J)]^j(u_{-j} - v_{-j}).$$

The above is a power series of the form

$$f(\theta) = 2\mathbf{w}_1 \sin^2(2J)(1-\epsilon)\sum_{j=0}^{\infty}\theta^j(u_{-j} - v_{-j}),$$

where $f(\theta)$ has a nonzero radius of convergence and is nonconstant since $\theta = (1-\epsilon)\cos^2(2J) \in (0, 1-\epsilon)$ and $(1-\epsilon)\sin^2(2J) \in (0, 1-\epsilon)$. Furthermore, since we assume that $u_0 = v_0$, we have $f(0) = 0$. Invoking Theorem 3.2 of Ref. [45], there exists $\beta > 0$ such that $f(\theta) \neq 0$ for all $|\theta| \leq \beta, \theta \neq 0$. This concludes the proof for the separation of points. The universality of $\mathcal{M}$ now follows from the Stone-Weierstrass theorem.

APPENDIX B: INVARIANCE UNDER TIME-INVARIANT READOUT ERROR

The QR outputs are invariant under time-invariant readout error whenever a linear readout function is used. That is, when $R = 1$ in Eq. (3), the QR predicted outputs $\overline{y}_l$ remain unchanged under time-invariant readout error. Let $\mathcal{B} = \{|i\rangle\}$ be the computational basis for an n-qubit system, with $i = 1, \ldots, 2^n$. The readout error is characterized by a measurement calibration matrix A whose i,jth element $A_{i,j} = \mathrm{Pr}(i \mid j)$ is the probability of measuring the state $|i\rangle \in \mathcal{B}$ given that the state is prepared in the state $|j\rangle \in \mathcal{B}$.

We employ the readout error correction method described in Ref. [4]. For an n-qubit QR, at each time step l, we execute 2^n calibration circuits with each circuit initialized in one of the 2^n computational basis elements. The

outcomes are used to create the measurement calibration matrix A_l. The readout error at time step l is corrected by applying the pseudoinverse of A_l to the measured outcomes from the experiments.

For all experiments, the measurement outcomes are stored as the count of measuring each basis elements in $\mathcal{B}$. Let $\mathbf{v}_l = (\mathbf{v}_l^1, \ldots, \mathbf{v}_l^{2^n}, 1)$, where $\mathbf{v}_l^i$ is the count of measuring $|i\rangle \in \mathcal{B}$ at time step l. Let $\mathbf{z}_l = (\overline{\langle Z^{(1)}\rangle}_l, \ldots, \overline{\langle Z^{(n)}\rangle}_l, 1)$, where $\overline{\langle Z^{(i)}\rangle}_l$ is the finite-sampled approximation of $\langle Z^{(i)}\rangle_l$ for $i = 1, \ldots, n$. Then we have $\mathbf{z}_l = \mathbf{v}_l B$, where B is a linear transformation. After applying the readout error correction, we have $\mathbf{z}_l' = \mathbf{v}_l A_l^+ B$, where A_l^+ is the pseudoinverse of A_l. To optimize the readout function parameters $\mathbf{w}$, collect all measurement data in a matrix $\mathbf{v} = (\mathbf{v}_1^\top, \ldots, \mathbf{v}_L^\top)^\top$ so that $\mathbf{z} = (\mathbf{z}_1^\top, \ldots, \mathbf{z}_L^\top)^\top = \mathbf{v}B$, where L is the sequence length. The linear output of the quantum reservoir computer is $\overline{\mathbf{y}} = \mathbf{v}B\mathbf{w}$, where $\mathbf{w}$ includes the bias term $\mathbf{w}_c$. Append a corresponding row and column to $A_l^\dagger$ to account for the bias term. Suppose that the readout error is time invariant. Then $A^+ = A_l^+$ for $l = 1, \ldots, L$. The quantum reservoir computer output after readout error correction is $\overline{\mathbf{y}'} = \mathbf{v}A^+ B\mathbf{w}'$. Assume that A^+ has all rows linearly independent. Then ordinary least squares yields $B\mathbf{w}' = AB\mathbf{w}$. Now given test data with readout error correction, $\mathbf{v}_{\text{test}}A^+ B\mathbf{w}' = \mathbf{v}_{\text{test}}A^+ AB\mathbf{w} = \mathbf{v}_{\text{test}}B\mathbf{w}$. Therefore, the QR predicted outputs are invariant under time-invariant readout error.

APPENDIX C: EFFICIENT IMPLEMENTATIONS OF A SUBCLASS ON GATE-MODEL QUANTUM COMPUTERS

We detail the second (more efficient) implementation scheme described in Sec. V and show how the QR's convergence property leads to more efficient versions of both schemes in Sec. V. As in Sec. V, we consider QRCs with a single subsystem [$N = 1$ in Eq. (1)] and drop the subsystem index k in Eq. (2). For multiple subsystems, the efficient implementation is simply applied on each subsystem.

If qubit reset is available, we can implement the second scheme in Sec. V based on QND measurements [38]. In this scheme, to estimate $\langle Z^{(i)}\rangle_l$, we no longer need to reinitialize and rerun the N_m circuits from time 1. Instead, for each N_m circuit, we perform a QND measurement of $Z^{(i)}$ at time l, continue running the circuit until the next QND measurement, and so forth. QND measurements ensure the information encoded in ρ_l is retained from one timestep to the next. To process a length-L input sequence, each N_m circuit is run S shots so that the average of $N_m S$ measurements at time l estimates $\langle Z^{(i)}\rangle_l$. That is, this scheme needs $N_m SL$ applications of $T(u_l)$, but only $N_m S$ circuit runs compared to $N_m LS$ runs in the first scheme (see Sec. V). This presents a substantial saving as the number of circuit runs is independent of the input sequence length L.
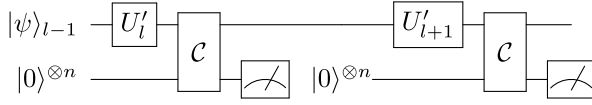
FIG. 6. Quantum circuit implementing the QND measurements by coupling ancilla qubits $|0\rangle^{\otimes n}$ with the QR system qubits $|\psi\rangle_{l-1}$.

To explain QND measurements, we first show that direct measurement of Z on a "system" qubit is equivalent to coupling the qubit with an ancilla qubit via CNOT and measuring Z_a, the Pauli Z operator acting on the ancilla qubit "a" [33]. To see this, let $|\psi\rangle_{\text{sys}} = \alpha|0\rangle_{\text{sys}} + \beta|1\rangle_{\text{sys}}$ be the state of the system qubit. Prepare the ancilla qubit at the ground state $|0\rangle_a$. We write

$$\text{CNOT} = |0\rangle\langle 0|_{\text{sys}} \otimes I_a + |1\rangle\langle 1|_{\text{sys}} \otimes X_a,$$

where I_a and X_a are the identity and Pauli X operators acting on the ancilla qubit. The system and ancilla state after applying CNOT is

$$|\Psi\rangle = \text{CNOT}|\psi\rangle_{\text{sys}} \otimes |0\rangle_a = \alpha|00\rangle + \beta|11\rangle.$$

Measurement of Z_a on the ancilla qubit is described by the projectors $P_+ = I_{\text{sys}} \otimes |0\rangle\langle 0|_a$ and $P_- = I_{\text{sys}} \otimes |1\rangle\langle 1|_a$. Therefore, the probabilities and postmeasurement system states are

$$\text{Pr}(+) = \langle\Psi|P_+|\Psi\rangle = |\alpha|^2,$$
$$\frac{\text{Tr}_a(P_+|\Psi\rangle\langle\Psi|P_+)}{\text{Pr}(+)} = |0\rangle\langle 0|_{\text{sys}},$$
$$\text{Pr}(-) = \langle\Psi|P_-|\Psi\rangle = |\beta|^2,$$
$$\frac{\text{Tr}_a(P_-|\Psi\rangle\langle\Psi|P_-)}{\text{Pr}(-)} = |1\rangle\langle 1|_{\text{sys}},$$

where $\text{Tr}_a(\cdot)$ is the partial trace over the ancilla qubit.

Now for an n-qubit QR, we associate each system qubit in the QR with its ancilla qubit. All n ancilla qubits are prepared in the ground state. Suppose that, when restricted to pure state preparation, we have drawn N_m circuits using Monte Carlo sampling. For each of the N_m circuits and each time step l, we apply the aforementioned ancilla-coupled measurement of $Z^{(i)}$ for each system qubit in the QR. After measuring the n ancilla qubits, we reset and reprepare them in the ground state for measurements at the next time $l+1$; see Fig. 6.

In Fig. 6, $|\psi\rangle_{l-1}$ denotes the state of the system (QR) qubits and $|0\rangle^{\otimes n}$ denotes the ancilla qubits initialized in the ground state. Here we have grouped the system and ancilla qubits and represent them using single wires. The unitary operator U'_l is U_0 or U_1 with probabilities $(1-\epsilon)u_l$ and $(1-\epsilon)(1-u_l)$, and $U(l) = U'_l\mathcal{C}$, where $\mathcal{C}$ is a product of n CNOT gates each acting on the ith system-ancilla qubit pair.

Measuring $Z_a^{(i)}$ on the ith ancilla qubit and resetting it at each time step $l = 1, \ldots, L$ is equivalent to having L ancilla qubits associated to the ith system qubit and measuring $Z_{a,l}^{(i)}$ (i.e., the lth ancilla qubit associated to the ith system qubit). The resulting QR dynamics are

$$T(u_l)\rho_{l-1} = (1-\epsilon)[u_lT_0 + (1-u_l)T_1]\rho_{l-1} + \epsilon\sigma,$$

where $T_j(\rho_{l-1}) = \text{Tr}_A[U_j\mathcal{C}\rho_{l-1} \otimes (|0\rangle\langle 0|)^{\otimes n}\mathcal{C}^\dagger U_j^\dagger]$ for $j = 0, 1$, and $\text{Tr}_A(\cdot)$ is the partial trace over all n ancilla qubits denoted by "A."

We now show that the measured observables $Z_{a,l}^{(i)}$ commute at different times as required by QND. More generally, we show that $Z_{a,l} = \bigotimes_{i=1}^n O_{a,l}^{(i)}$ ($l = 1, \ldots, L$), where, for each i, we have $O_{a,l}^{(i)} = I^{(i)}$ (the identity operator on the ith qubit) or $O_{a,l}^{(i)} = Z_{a,l}^{(i)}$, are QND observables. Firstly, we have the commutator $[Z_{a,k}, Z_{a,j}] = 0$ for all $k, j = 1, \ldots, L$. Denote the evolved observables in the Heisenberg picture by

$$Z_a(l) = U(1)^\dagger \cdots U(l)^\dagger Z_{a,l}U(l) \cdots U(1) = U_{l:1}^\dagger Z_{a,l}U_{l:1},$$

where $U_{l:1} = U(l) \cdots U(1)$. For $k, j = 1, \ldots, L$ with $j < k$, we have

$$[Z_a(j), Z_a(k)]$$
$$= U_{j:1}^\dagger Z_{a,j}U_{j:1}U_{k:1}^\dagger Z_{a,k}U_{k:1} - U_{k:1}^\dagger Z_{a,k}U_{k:1}U_{j:1}^\dagger Z_{a,j}U_{j:1}$$
$$= U_{j:1}^\dagger Z_{a,j}U_{k:j+1}^\dagger Z_{a,k}U_{k:1} - U_{k:1}^\dagger Z_{a,k}U_{k:j+1}Z_{a,j}U_{j:1}$$
$$= U_{k:1}^\dagger[Z_{a,j}, Z_{a,k}]U_{k:1}$$
$$= 0,$$

where in the second-to-last equality we have used the fact that $Z_{a,j}$ commutes with the future unitary operations $U_{k:j+1}$. If $j > k$, applying the same argument as above shows that $[Z_a(j), Z_a(k)] = -[Z_a(k), Z_a(j)] = 0$. The commutativity of $Z_a(j)$ and $Z_a(k)$ for all $j, k \geq 1$ means that the sequence $\{Z_a(j), j = 1, 2, \ldots\}$ has a joint probability distribution and constitutes a classical stochastic process. QND measurements on the sequence gives a realization of this stochastic process.

The QR's convergence property (see Appendix A 1) leads to more efficient versions of both schemes in Sec. V. Let M, ρ_l, ρ_{l-M}, and $\tilde{\rho}_l$ be as given in Sec. V. By the convergence property [Eq. (A1)], we have

$$\|\rho_l - \tilde{\rho}_l\|_1 \leq (1-\epsilon)^M\|\rho_{l-M} - (|0\rangle\langle 0|)^{\otimes n}\|_1 \leq 2(1-\epsilon)^M. \tag{C1}$$

The difference between ρ_l and $\tilde{\rho}_l$ can be made negligible by choosing M appropriately based on ϵ. If we perform repeated measurements on ρ_l and $\tilde{\rho}_l$, then the finite-sample

estimates of $\langle \widetilde{Z^{(i)}} \rangle_l = \mathrm{Tr}(\tilde{\rho}_l Z^{(i)})$ and $\langle Z^{(i)} \rangle_l = \mathrm{Tr}(\rho_l Z^{(i)})$ will also be close; see Appendix D. Using the convergence property, the first scheme in Sec. V requires $N_m SL$ circuit runs but only $N_m SLM$ applications of $T(u_l)$. When $L > M$, a substantial saving in the number of applications of T can be obtained (for timesteps $l > M$) compared to the previous quadratic dependence on L. The second scheme now only requires at most $N_m SM$ applications of $T(u_l)$ and $N_m S$ circuit runs, both are *independent* of the input sequence length L. This provides a path for fast and large scale temporal processing using QRs.

APPENDIX D: MONTE CARLO ESTIMATION

For all schemes described in Sec. V, we can set $S = 1$ and run N_m Monte Carlo sampled circuits (possibly in parallel if many copies of the same hardware are available) for a sufficiently large N_m. We show that the average of all N_m measurements at time l estimates $\langle Z^{(i)} \rangle_l$ and its variance vanishes as N_m tends to infinity.

First consider estimating $\langle Z^{(i)} \rangle_l$ by reinitializing each N_m circuit in $|0\rangle^{\otimes n}$ and rerunning them from time 1 to time l according to inputs $\{u_1, \ldots, u_l\}$. Recall that

$$\langle Z^{(i)} \rangle_l = \mathrm{Tr}(Z^{(i)} \rho_l) = \mathrm{Tr}[Z^{(i)} T(u_l) \cdots T(u_1)(|0\rangle \langle 0|)^{\otimes n}],$$

where $T(u_k)$ is the input-dependent CPTP map defined in Eq. (1) for $k = 1, \ldots, l$. Define independent discrete-valued random variables X_k such that

$$\Pr(X_k = 0) = (1 - \epsilon) u_k,$$
$$\Pr(X_k = 1) = (1 - \epsilon)(1 - u_k),$$
$$\Pr(X_k = 2) = \epsilon.$$

To implement the QR, for each time k, we independently sample N_m random variables $X_{k,j}$ ($j = 1, \ldots, N_m$) from the same distribution as X_k. Define

$$T_x = \begin{cases} T_0 & \text{if } x = 0, \\ T_1 & \text{if } x = 1, \\ K_\sigma & \text{if } x = 2, \end{cases}$$

where $K_\sigma(\rho) = \sigma$ is a constant CPTP map that sends any density operator ρ to the constant density operator σ in Eq. (2). The random CPTP maps $T_{X_{k,j}}$ follow the same distribution as $X_{k,j}$ and are independent for each k and j. Furthermore, $\mathbb{E}[T_{X_{k,j}}] = T(u_k)$.

For the jth circuit, we implement a sequence of (random) CPTP maps $T_{X_{l,j}} \cdots T_{X_{1,j}}$ so that at time l, the (random) QR state is

$$\rho^{\mathbf{X}_{l,j}} = T_{X_{l,j}} \cdots T_{X_{1,j}} (|0\rangle \langle 0|)^{\otimes n},$$

where $\mathbf{X}_{l,j} = (X_{1,j}, \ldots, X_{l,j})$. For each jth circuit, we measure $Z^{(i)}$ and denote its random outcome by $\overline{Z^{(i)}}_{l,j}$. Note that, for $j = 1, \ldots, N_m$, the $\overline{Z^{(i)}}_{l,j}$ are independent (but not necessarily identically distributed) random variables taking values ± 1 (eigenvalues of $Z^{(i)}$) with conditional probabilities (conditional on the random variables $\mathbf{X}_{l,j}$)

$$\Pr(\overline{Z^{(i)}}_{l,j} = z \mid \mathbf{X}_{l,j}) = \mathrm{Tr}(\rho^{\mathbf{X}_{l,j}} P_z^{(i)}), \qquad z = \pm 1,$$

where the $P_{\pm 1}^{(i)}$ are the projectors such that $Z^{(i)} = P_{+1}^{(i)} - P_{-1}^{(i)}$. Consider the average of all N_m measurement outcomes; by the law of total expectation,

$$\frac{1}{N_m} \sum_{j=1}^{N_m} \mathbb{E}[\overline{Z^{(i)}}_{l,j}]$$

$$= \frac{1}{N_m} \sum_{j=1}^{N_m} \mathbb{E}\{\mathbb{E}[\overline{Z^{(i)}}_{l,j} \mid \mathbf{X}_{l,j}]\}$$

$$= \frac{1}{N_m} \sum_{j=1}^{N_m} \mathbb{E}[\mathrm{Tr}(Z^{(i)} \rho^{\mathbf{X}_{l,j}})]$$

$$= \frac{1}{N_m} \sum_{j=1}^{N_m} \mathrm{Tr}\{Z^{(i)} \mathbb{E}[T_{X_{l,j}}] \cdots \mathbb{E}[T_{X_{1,j}}](|0\rangle \langle 0|)^{\otimes n}\}$$

$$= \frac{1}{N_m} \sum_{j=1}^{N_m} \mathrm{Tr}[Z^{(i)} T(u_l) \cdots T(u_1)(|0\rangle \langle 0|)^{\otimes n}]$$

$$= \mathrm{Tr}(\rho_l Z^{(i)})$$

$$= \langle Z^{(i)} \rangle_l.$$

Therefore, the finite-sample estimate is unbiased. Moreover, using the fact that

$$\mathbb{E}[(\overline{Z^{(i)}}_{l,j})^2] = \sum_{z=\pm 1} z^2 \Pr(\overline{Z^{(i)}}_{l,j} = z) = 1,$$

the variance of the average of N_m measurements is

$$\mathrm{Var}\left[\frac{1}{N_m} \sum_{j=1}^{N_m} \overline{Z^{(i)}}_{l,j}\right] = \frac{1}{N_m^2} \sum_{j=1}^{N_m} \mathrm{Var}[\overline{Z^{(i)}}_{l,j}]$$

$$= \frac{1}{N_m}(1 - \langle Z^{(i)} \rangle_l^2).$$

Using the convergence property to estimate $\langle Z^{(i)} \rangle_l$ for a sufficiently large l (that depends on ϵ), we reinitialize N_m
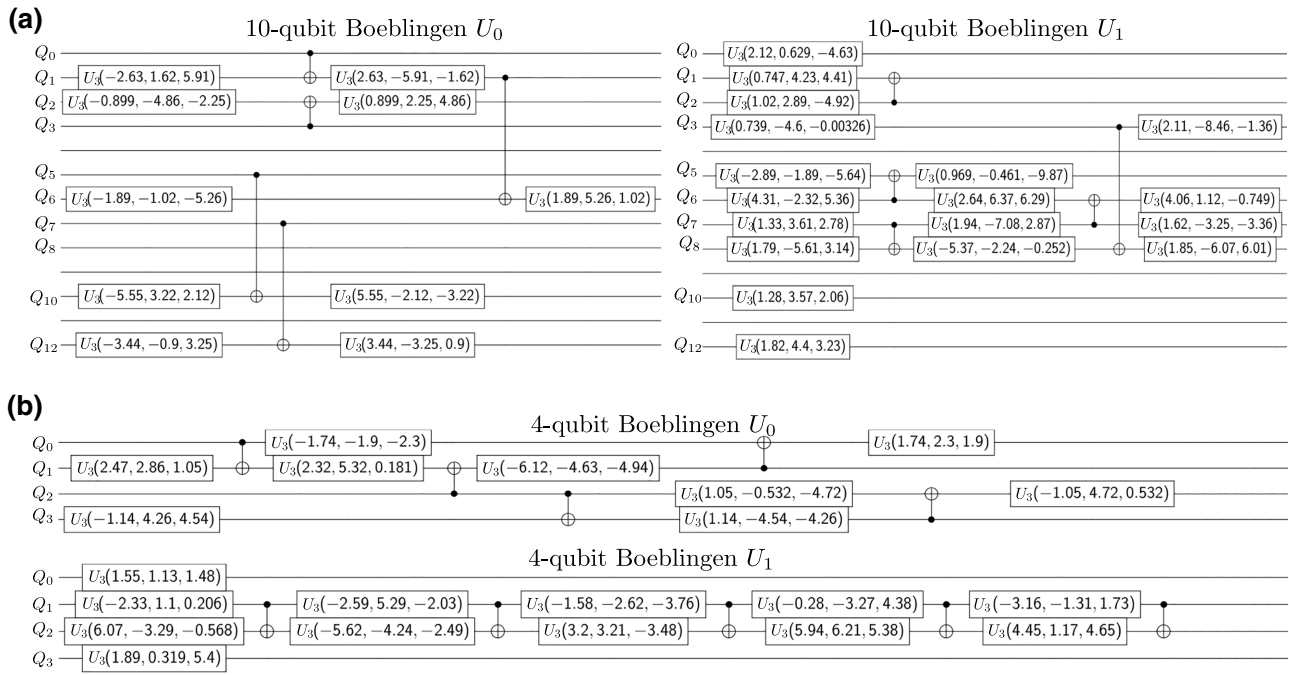
FIG. 7. Quantum circuits for the (a) 10-qubit QR and (b) 4-qubit QR on the Boeblingen device.

circuits at time $l - M$ and run the circuits according to inputs $\{u_{l-M+1}, \ldots, u_l\}$. Let $\langle \widetilde{Z^{(i)}} \rangle_l = \mathrm{Tr}(Z^{(i)} \tilde{\rho}_l)$, where

$$\tilde{\rho}_l = T(u_l) \cdots T(u_{l-M+1})(|0\rangle\langle 0|)^{\otimes n}.$$

In this setting, for the j th circuit, we implement a sequence of (random) CPTP maps $T_{X_{l,j}} \cdots T_{X_{l-M+1,j}}$ so that the (random) QR state at time l is

$$\rho^{\widetilde{\mathbf{X}}_{l,j}} = T_{X_{l,j}} \cdots T_{X_{l-M+1,j}}(|0\rangle\langle 0|)^{\otimes n},$$

where $\widetilde{\mathbf{X}}_{l,j} = (X_{l-M+1,j}, \ldots, X_{l,j})$. Let $\widetilde{Z^{(i)}}_{l,j}$ be the random outcome of measuring $Z^{(i)}$. The conditional probabilities are

$$\mathrm{Pr}(\widetilde{Z^{(i)}} = z \mid \widetilde{\mathbf{X}}_{l,j}) = \mathrm{Tr}(\rho^{\widetilde{\mathbf{X}}_{l,j}} P_z^{(i)}), \qquad z = \pm 1.$$

A similar argument as above shows that the average of all N_m measurements satisfies

$$\mathbb{E}\left[\frac{1}{N_m}\sum_{j=1}^{N_m} \widetilde{Z^{(i)}}_{l,j}\right] = \langle \widetilde{Z^{(i)}} \rangle_l,$$

$$\mathrm{Var}\left[\frac{1}{N_m}\sum_{j=1}^{N_m} \widetilde{Z^{(i)}}_{l,j}\right] = \frac{1}{N_m}(1 - \langle \widetilde{Z^{(i)}} \rangle_l^2).$$

The convergence property and Eq. (C1) ensure that the bias (in mean) vanishes exponentially fast,

$$\left|\mathbb{E}\left[\frac{1}{N_m}\sum_{j=1}^{N_m} \widetilde{Z^{(i)}}_{l,j}\right] - \langle Z^{(i)} \rangle_l\right| = |\mathrm{Tr}[Z^{(i)}(\tilde{\rho}_l - \rho_l)]|$$

$$\leq \|\tilde{\rho}_l - \rho_l\|_1$$

$$\leq 2(1 - \epsilon)^M,$$

where we have used the fact that, for any Hermitian matrix A, $|\mathrm{Tr}(Z^{(i)} A)| \leq \sigma_{\max}(Z^{(i)})\|A\|_1$, with $\sigma_{\max}(Z^{(i)}) =$
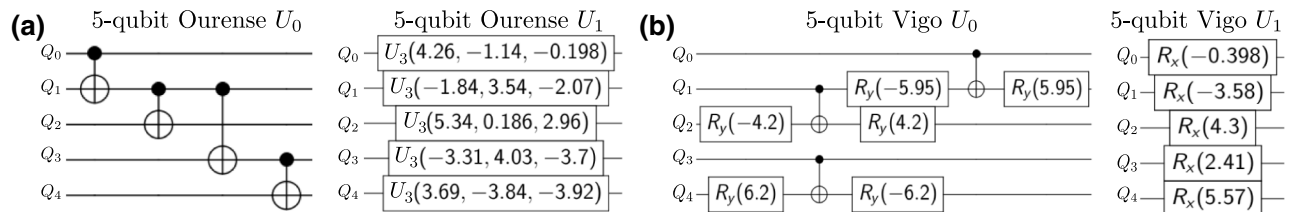

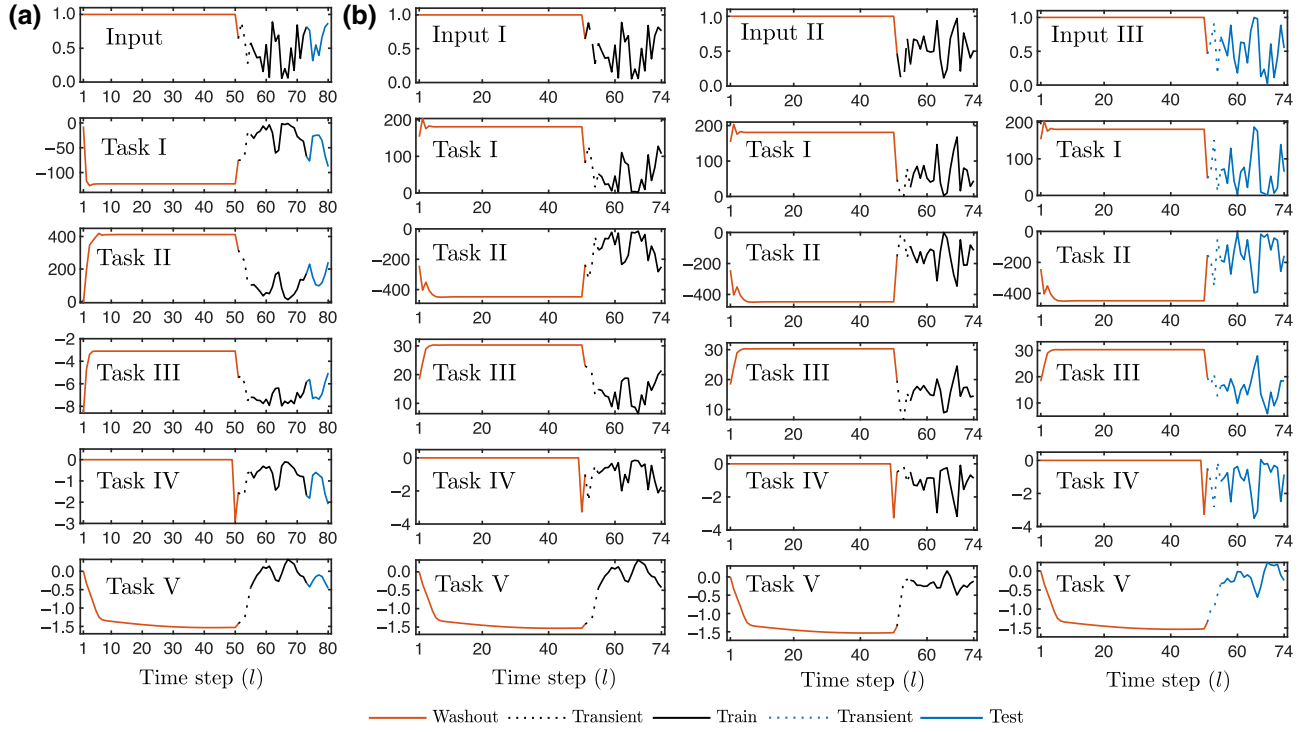FIG. 8. Quantum circuits for the (a) 5-qubit Ourense QR and (b) 5-qubit Vigo QR.

FIG. 9. Full washout, train, and test input-output sequences for (a) the multistep ahead prediction problem and (b) the map emulation problem. The first rows in (a) and (b) show the input sequences.

1 denoting the maximum singular value of $Z^{(i)}$. This shows that the bias can be exponentially suppressed by choosing M appropriately based on ϵ, so that the estimates of $\langle \widetilde{Z^{(i)}} \rangle_l$ and $\langle Z^{(i)} \rangle_l$ are also close.

APPENDIX E: EXPERIMENTAL AND NUMERICAL DETAILS

1. Nonlinear temporal processing tasks

We give detailed descriptions of the five nonlinear temporal processing tasks. Tasks I and II are governed by linear reservoirs with polynomial readout [27,29], described by

$$\mathbf{x}_l = A\mathbf{x}_{l-1} + c u_l, \qquad y_l = h(\mathbf{x}_l),$$

where $A \in \mathbb{R}^{2000 \times 2000}$ and $c \in \mathbb{R}^{2000}$. To have short-term or fading memory, we rescale the maximum singular value $\sigma_{\max}(A) = 0.5$ for task I and $\sigma_{\max}(A) = 0.99$ for task II, meaning that task II retains the initial condition and past inputs for a longer time duration. The sparsity of A determines the pairwise correlation between reservoir state elements. We set A to be a full (dense) matrix for task I and 95% sparse for task II. The readout function h is a degree-two polynomial in the state elements. Task III is a recently proposed classical reservoir computing model that achieves good performance in chaotic system

modeling [27], described by

$$\mathbf{x}_l = p(u_l)\mathbf{x}_{l-1} + q(u_l), \qquad y_l = \mathbf{w}^T \mathbf{x}_l,$$

where $p(u_l) = \sum_{j=0}^{4} A_j u_l^j$ and $q(u_l) = \sum_{j=0}^{2} B_j u_l^j$ are matrix-valued polynomials in the input u_l, $A_j \in \mathbb{R}^{700 \times 700} \bigoplus \mathbb{R}^{700 \times 700}$, and $B_j \in \mathbb{R}^{700 \times 1} \bigoplus \mathbb{R}^{700 \times 1}$. For task III, We rescale $\sigma_{\max}(A_j) < \frac{1}{3}$ for all j so that it exhibits short-term memory. Task IV is a Volterra series with kernel order 5 and memory 2, commonly applied to model responses of nonlinear systems in control engineering [29]:

$$y_l = \mathbf{w}_c + \sum_{i=1}^{5} \sum_{j_1, \dots, j_i = 0}^{2} \mathbf{w}_i^{j_1, \dots, j_i} \prod_{k=1}^{i} u_{l-j_k}.$$

For the first three tasks, elements of A, A_j, B, and $\mathbf{w}$ are uniformly randomly sampled from $[-1, 1]$. The constant c and coefficients of the readout function h are also sampled from the same distribution. The same applies to the kernel coefficients $\mathbf{w}_i^{j_1, \dots, j_i}$ and $\mathbf{w}_c$ in task IV.

Task V is a missile moving with a constant velocity in the horizontal plane, a continuous-time long-term memory nonlinear map [46] described by

$$\dot{\mathbf{x}}_1 = \mathbf{x}_2 - 0.1\cos(\mathbf{x}_1)(5\mathbf{x}_1 - 4\mathbf{x}_1^3 + \mathbf{x}_1^5) - 0.5\cos(\mathbf{x}_1)u,$$

$$\dot{\mathbf{x}}_2 = -65\mathbf{x}_1 + 50\mathbf{x}_1^3 - 15\mathbf{x}_1^5 - \mathbf{x}_2 - 100u,$$
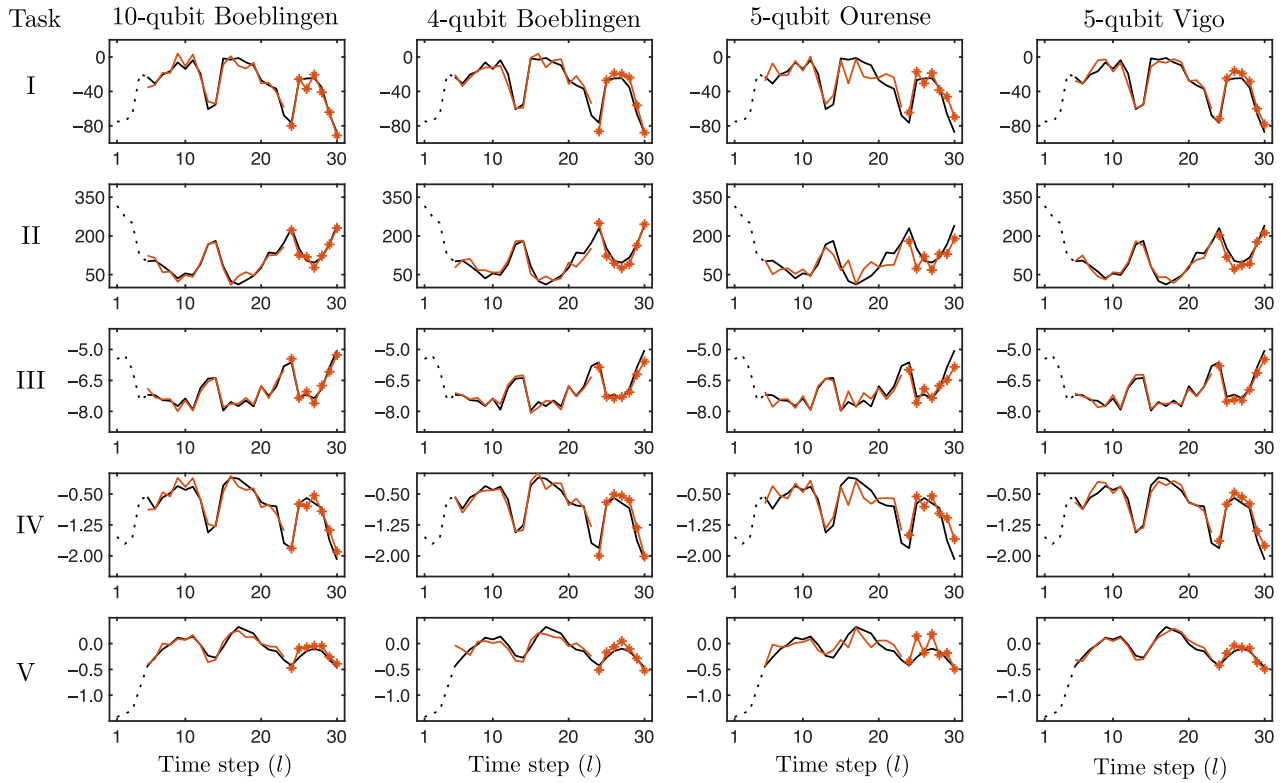
FIG. 10. The full target output sequences, the train and test output sequences of the four QRs for each task on the multistep ahead prediction problem. Each column corresponds to each n-qubit QR output and each row corresponds to each task.

with $y = \mathbf{x}_2$. The missile dynamics is simulated using the $(4, 5)$ Runge-Kutta formula in MATLAB, with a sampling time of $\tau = 1/80$ for 1 s.

2. Quantum circuits for QRs

We detail the circuits implementing the QR dynamics in our proof-of-principle experiments presented in Sec. VI. The quantum circuits for the 4-qubit and 10-qubit Boeblingen QRs are shown in Fig. 7. The quantum circuits for the 5-qubit Ourense and 5-qubit Vigo QRs are shown in Fig. 8.

3. Full input-output sequential data

Since we bypass the washout for QRs by initializing them in the state $|0\rangle^n$, this is equivalent to washing out their initial conditions with a length L_w constant input sequence $u_l = 1$. The same washout has been applied to all nonlinear tasks. We have checked that $L_w = 50$ is enough for all tasks to reach steady states given the same initialization $\mathbf{x}_0 = 0$. Particular caution has been taken to washout task IV, in which we set $u_l = 1$ for $l = -2, -1$. For each target map, we discard the first four input-output sequence data points, and the corresponding QR experimental data, to remove the transitory output response due to the change in input statistics. In Fig. 9, we show the full washout, train, and test input-output target sequences for both the

multistep ahead prediction and the map emulation problems. In Fig. 10 we plot the full target output sequences, the train and test QR outputs on the multistep ahead prediction problem. In Fig. 11 we plot the full target output sequences, the train and test QR outputs on the map emulation problem. In all figures, the transitory responses are indicated by dotted lines.

4. Measurement and simulation data

We simulate the four QRs using the IBM Qiskit simulator under ideal and noisy conditions. The noise models used are obtained from the device calibration data. We fetch the updated device calibration data each time a job is executed on the hardware. The circuits simulated are the same as the circuits employed for the experiments and so is the number of shots. For the multistep ahead prediction problem, the 10-qubit Boeblingen QR experienced a significant deviation from simulated results on qubits $Q = 1, 8$ (see Fig. 12), resulting in larger $\mathcal{E} = 0.26, 0.29, 0.068, 0.15, 6.1$ for the four tasks. After setting the readout parameters $\mathbf{w}_1 = \mathbf{w}_8 = 0$ for $Q = 1, 8$, this issue was circumvented at the cost of using a fewer number of computational features. The resulting 10-qubit Boeblingen QR still achieves performance improvement over other QRs with a smaller number of qubits on the multistep
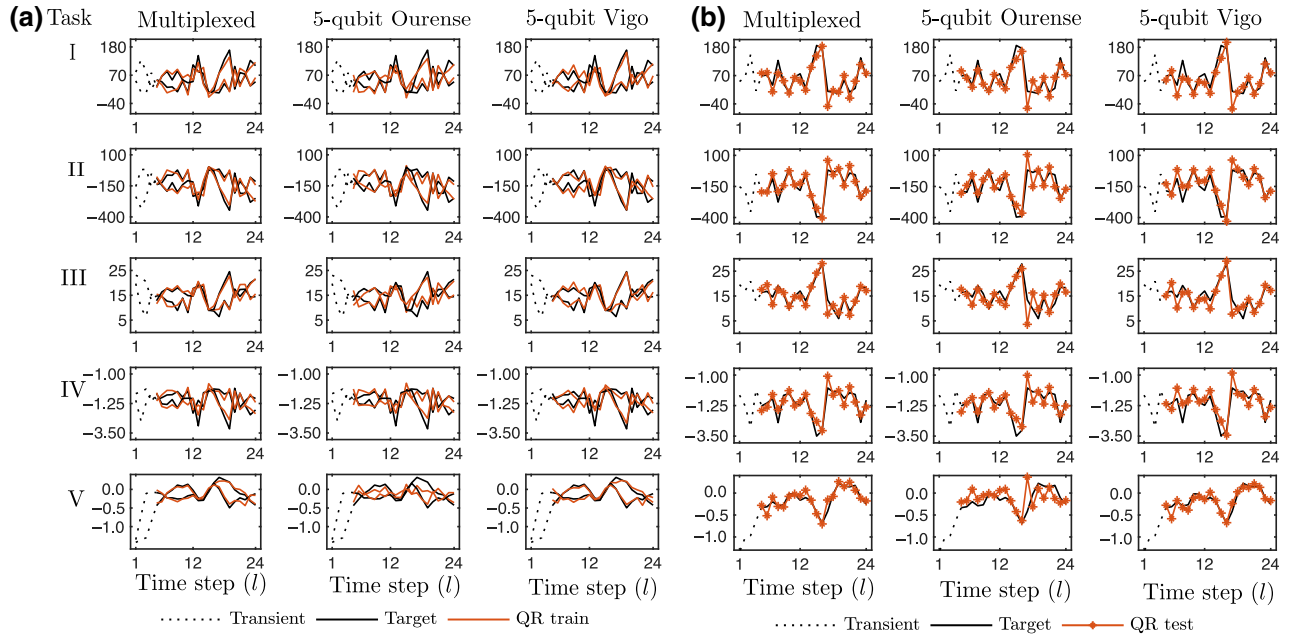
FIG. 11. The full target output sequences, the train and test output sequences of the QRs for each task on the map emulation problem. (a) The two train output sequences and (b) the test output sequence. The columns (from left to right) correspond to the multiplexed 5-qubit QRs, 5-qubit Ourense QR, and the 5-qubit Vigo QR. Each row corresponds to each task.

ahead prediction problem in the first three tasks. A time-invariant readout error in qubit i linearly transforms the expectation $\langle Z^{(i)} \rangle_l$. The QR predicted outputs are invariant under time-invariant readout errors when using linear regression to optimize $\mathbf{w}, \mathbf{w}_c$ as derived in Appendix B. However, for the 10-qubit Boeblingen QR, the deviations in qubits $Q = 1, 8$ are time varying. On the other hand, the 5-qubit Vigo device experienced almost time-invariant deviations in qubit $Q = 0$, as shown in Figs. 12 and 13, but this does not affect the performance of this QR noticeably. The experimental results of the 5-qubit Ourense QR follow the noisy simulation results closely. For the map emulation problem, the experimental results of both 5-qubit QRs follow the simulated results closely, with an almost time-invariant shift in $Q = 0$ for the 5-qubit Vigo QR.

5. Hardware specifications

The experiments are conducted on the IBM 20-qubit Boeblingen (version 1.0.0), 5-qubit Ourense (version 1.0.0), and 5-qubit Vigo (version 1.0.0) superconducting quantum processors [31]. The gate duration for an arbitrary single-qubit rotation gate U_3 [40] is $\tau_{U_3} \approx 71.1$ ns for all qubits, whereas the CNOT gate durations differ for different qubits.

See Fig. 7 for the 4-qubit and 10-qubit Boeblingen QR quantum circuits. The circuits are chosen such that both QRs have the same number of layers in U_0 and U_1. In this setting, the maximum duration of a circuit executed on the Boeblingen device is the same for both QRs. As stated in

the main text, the chosen qubits for the 4-qubit QR and the 10-qubit QR on the Boeblingen device are $Q = 0, 1, 2, 3$ and $Q = 0, 1, 2, 3, 5, 6, 7, 8, 10, 12$. These qubits are chosen due to their longer coherence times, shorter CNOT gate durations, and smaller gate and readout errors. During the experiment, the maximum readout error is 10^{-2} and the maximum U_3 gate error implemented is 10^{-3}. The maximum CNOT gate error implemented is 4.3×10^{-2} and the maximum CNOT gate duration is $\tau_{\mathrm{CNOT}} \approx 427$ ns. We assume that commuting gates can be executed in parallel. We choose $N_0 = N_1 = 5$ numbers of layers for U_0 and U_1 in the 4-qubit and 10-qubit Boeblingen QRs. The maximum length of any input sequence (including the transient) for the multistep ahead prediction and the map emulation problems is $L = 30$. Therefore, the maximum numbers of U_3 gate executions and CNOT gate executions is $5L = 5 \times 30 = 150$. The maximum duration of a circuit executed on the Boeblingen device is $150 \times (\tau_{U_3} + \tau_{\mathrm{CNOT}}) \approx 150 \times (71.1 + 427) = 74.7 \ \mu\mathrm{s}$, within the coherence times (T_1, T_2) for most qubits chosen.

In Fig. 8 we show the quantum circuits for the 5-qubit Ourense and 5-qubit Vigo QRs. Owing to the more restricted qubit couplings in these 5-qubit devices, the circuits for the 5-qubit QRs are simpler than those of the 4-qubit and 10-qubit Boeblingen QRs. To combine different computational features for the spatial multiplexing technique, we choose circuits that are sufficiently different for these two 5-qubit QRs. In particular, the 5-qubit Vigo QR consists of single-qubit rotational Y gates in U_0 and single-qubit rotational X gates in U_1.
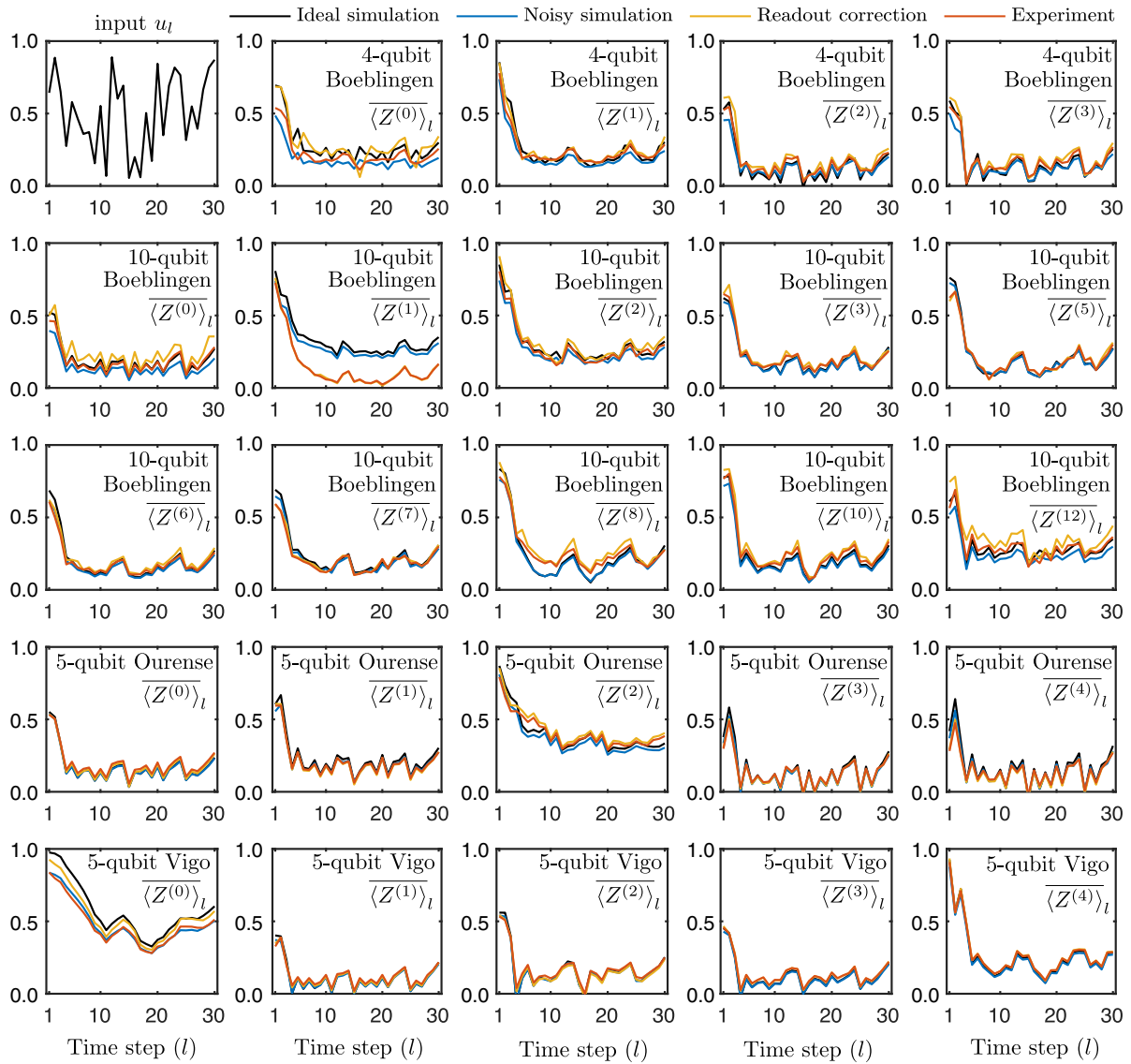
FIG. 12. Input sequence, and experimental and simulation results for each qubit of the four QRs at each time step $l = 1, \ldots, 30$ for the multistep ahead prediction problem.

On the other hand, the 5-qubit Ourense QR uses arbitrary single-qubit rotational gates U_3 only in the circuit implementing U_1.

The 5-qubit Ourense device achieves the same order of magnitude in readout errors, coherence times, and CNOT gate durations as the 20-qubit Boeblingen device, but lower CNOT gate errors. For the Ourense device, the maximum U_3 gate error and readout error implemented are 0.9×10^{-3} and 4.1×10^{-2}, and the maximum CNOT gate error implemented is 8×10^{-3}, a lower error compared to the Boeblingen device. The maximum CNOT gate duration implemented is $\tau_{\text{CNOT}} \approx 576$ ns. For the 5-qubit Ourense QR, the circuit implementing U_0 is longer than that for U_1. The U_0 circuit consists of four CNOT gates, and the maximum duration of a circuit executed for the 5-qubit Ourense

QR is $4L \times \tau_{\text{CNOT}} \approx 70$ μs, also within the coherence limits of most qubits.

The 5-qubit Vigo device is similar to the 5-qubit Ourense device. They have the same qubit couplings and share similar noise profile and hardware specifications. Rotational X and Y gates are used on this device, with gate duration $\tau = 35.5$ ns. The maximum single-qubit gate error implemented is 0.8×10^{-3} and the maximum readout error implemented is 7.8×10^{-2}. The maximum CNOT gate error and gate duration implemented are 1.3×10^{-2} and $\tau_{\text{CNOT}} \approx 462.2$ ns, respectively. For this QR, U_0 is the longer circuit consisting of three layers of single-qubit rotation Y gates and two layers of CNOT gates. Therefore, the maximum duration of a circuit implemented is $(3\tau + 2\tau_{\text{CNOT}})L = (3 \times 35.5 + 2 \times 462.2) \times 30 \approx 31$ μs.
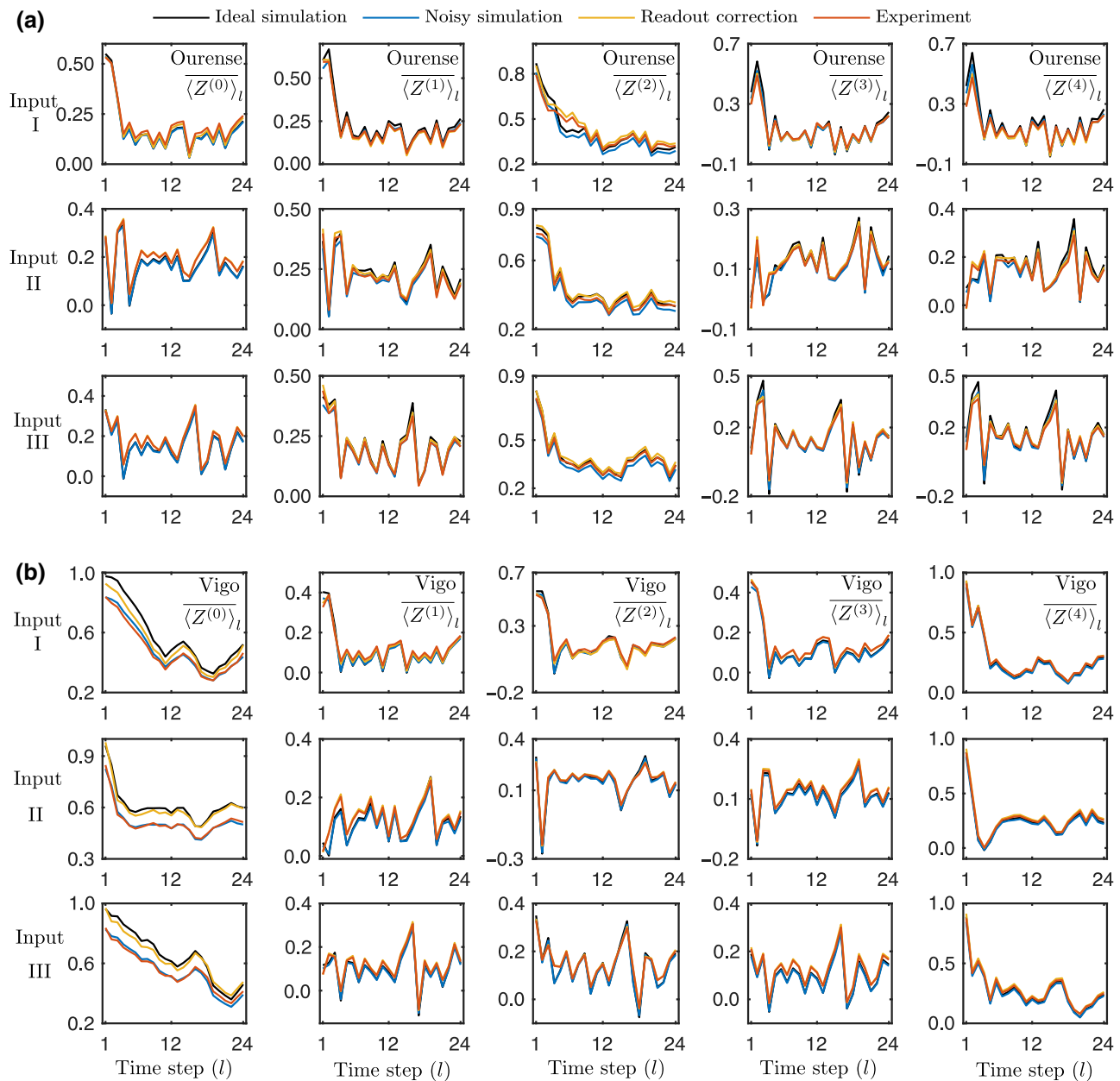
FIG. 13. Experimental and simulation results for each qubit $i = 0, \ldots, 4$ and each time step $l = 1, \ldots, 24$ for the map emulation problem. Three input sequences are used in this problem, labeled as inputs I, II, and III. Row i in a subfigure corresponds to the experimental data for the ith input sequence. Column j corresponds to the experimental data for the jth qubit. (a) The experimental data for the 5-qubit Ourense QR. (b) The experimental data for the 5-qubit Vigo QR.

[1] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, Quantum machine learning, Nature **549**, 195 (2017).

[2] C. Ciliberto, M. Herbster, A. D. Ialongo, M. Pontil, A. Rocchetto, S. Severini, and L. Wossnig, Quantum machine learning: A classical perspective, Proc. R. Soc. A: Math., Phys. Eng. Sci. **474**, 20170551 (2018).

[3] J. Preskill, Quantum computing in the NISQ era and beyond, Quantum **2**, 79 (2018).

[4] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, Nature **567**, 209 (2019).

[5] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow, and J. M. Gambetta, Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets, Nature **549**, 242 (2017).

[6] J. Miller and M. Hardt, in *Proceedings of the 2019 International Conference on Learning Representations (ICLR)* (Curran Associates, New York, 2019), arXiv:1805.10369.

[7] J. Hanson and M. Raginsky, in *Proceedings of the 33rd Annual Conference on Neural Information Processing Systems (NeurIPS)* (OpenReview, 2019), p. 14071.

[8] M. Rigotti, O. Barak, M. R. Warden, X.-J. Wang, N. D. Daw, E. K. Miller, and S. Fusi, The importance of mixed selectivity in complex cognitive tasks, Nature **497,** 585 (2013).

[9] R. Pascanu, T. Mikolov, and Y. Bengio, in *International Conference on Machine Learning (ICML)* (Proceedings of Machine Learning Research, 2013), p. 1310.

[10] C. Du, F. Cai, M. A. Zidan, W. Ma, S. H. Lee, and W. D. Lu, Reservoir computing using dynamic memristors for temporal information processing, Nat. Commun. **8,** 2204 (2017).

[11] K. Vandoorne, P. Mechet, T. Van Vaerenbergh, M. Fiers, G. Morthier, D. Verstraeten, B. Schrauwen, J. Dambre, and P. Bienstman, Experimental demonstration of reservoir computing on a silicon photonics chip, Nat. Commun. **5,** 3541 (2014).

[12] Q. Vinckier, F. Duport, A. Smerieri, K. Vandoorne, P. Bienstman, M. Haelterman, and S. Massar, High-performance photonic reservoir computer based on a coherently driven passive cavity, Optica **2,** 438 (2015).

[13] J. Torrejon, M. Riou, F. A. Araujo, S. Tsunegi, G. Khalsa, D. Querlioz, P. Bortolotti, V. Cros, K. Yakushiji, A. Fukushima, H. Kubota, S. Yuasa, M. Stiles, and J. Grollier, Neuromorphic computing with nanoscale spintronic oscillators, Nature **547,** 428 (2017).

[14] L. Larger, A. Baylón-Fuentes, R. Martinenghi, V. S. Udaltsov, Y. K. Chembo, and M. Jacquot, High-Speed Photonic Reservoir Computing Using a Time-Delay-Based Architecture: Million Words per Second Classification, Phys. Rev. X **7,** 011015 (2017).

[15] L. Gonon, L. Grigoryeva, and J.-P. Ortega, Approximation bounds for random neural networks and reservoir systems, arXiv:2002.05933 (2020).

[16] J. Dambre, D. Verstraeten, B. Schrauwen, and S. Massar, Information processing capacity of dynamical systems, Sci. Rep. **2,** 1 (2012).

[17] L. Gonon, L. Grigoryeva, and J.-P. Ortega, Memory and forecasting capacities of nonlinear recurrent networks, arXiv:2004.11234 (2020).

[18] G. Tanaka, T. Yamane, J. B. Héroux, R. Nakane, N. Kanazawa, S. Takeda, H. Numata, D. Nakano, and A. Hirose, Recent advances in physical reservoir computing: A review, Neural. Netw.**115,** 100 (2019).

[19] K. Nakajima, Physical reservoir computing–an introductory perspective, Jpn. J. Appl. Phys. **59,** 060501 (2020).

[20] W. Maass, T. Natschläger, and H. Markram, Real-time computing without stable states: A new framework for neural computation based on perturbations, Neural Comput. **14,** 2531 (2002).

[21] K. Fujii and K. Nakajima, Harnessing Disordered-Ensemble Quantum Dynamics for Machine Learning, Phys. Rev. Appl. **8,** 024030 (2017).

[22] K. Nakajima, K. Fujii, M. Negoro, K. Mitarai, and M. Kitagawa, Boosting Computational Power through Spatial Multiplexing in Quantum Reservoir Computing, Phys. Rev. Appl. **11,** 034021 (2019).

[23] M. Negoro, K. Mitarai, K. Fujii, K. Nakajima, and M. Kitagawa, Machine learning with controllable quantum dynamics of a nuclear spin ensemble in a solid, arXiv:1806.10910 (2018).

[24] J. Chen and H. I. Nurdin, Learning nonlinear input–output maps with dissipative quantum systems, Quantum Inf. Process. **18,** 198 (2019).

[25] G. Cybenko, Approximation by superpositions of a sigmoidal function, Math. Control, Signals Syst. **2,** 303 (1989).

[26] K. Hornik, M. Stinchcombe, and H. White, Multilayer feedforward networks are universal approximators, Neural Netw. **2,** 359 (1989).

[27] L. Grigoryeva and J.-P. Ortega, Universal discrete-time reservoir computers with stochastic inputs and linear readouts using non-homogeneous state-affine systems, J. Mach. Learning Res. **19,** 892 (2018).

[28] L. Grigoryeva and J. Ortega, Echo state networks are universal, Neural Netw. **108,** 495 (2018).

[29] S. Boyd and L. Chua, Fading memory and the problem of approximating nonlinear operators with Volterra series, IEEE Trans. Circuits Syst. **32,** 1150 (1985).

[30] H. Jaeger and H. Haas, Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communications, Science **304,** 5667 (2004).

[31] IBM Quantum Experience, https://www.ibm.com/quantum-computing/ (2019).

[32] A. Pavlov, N. van de Wouw, and H. Nijmeijer, in *Control and Observer Design for Nonlinear Finite and Infinite Dimensional Systems*, edited by T. Meurer, K. Graichen, and E. D. Gilles, Lecture Notes in Control and Information Science Vol. 322 (Springer, Germany, 2005), p. 131.

[33] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, New York, 2010).

[34] J. Dieudonné, *Foundations of Modern Analysis* (Read Books Ltd, New York, 2013).

[35] Qiskit Aer, Device backend noise model simulations, https://github.com/Qiskit/qiskit-iqx-tutorials/blob/master/qiskit/advanced/aer/2˙device˙noise˙simulation.ipynb (2019).

[36] Qiskit Aer API Reference, https://qiskit.org/documentation/apidoc/aer.html (2019).

[37] J. Chen, H. I. Nurdin, and N. Yamamoto, in *Proceedings of the 2019 IEEE Conference on Decision and Control (CDC)* (IEEE, Piscataway, NJ, 2019), p. 401.

[38] V. B. Braginsky and F. Y. Khalili, *Quantum Measurement* (Cambridge University Press, New York, 1995).

[39] J. Pino, J. Dreiling, C. Figgatt, J. Gaebler, S. Moses, C. Baldwin, M. Foss-Feig, D. Hayes, K. Mayer, C. Ryan-Anderson, and B. Neyenhuis, Demonstration of the QCCD trapped-ion quantum computer architecture, arXiv:2003.01293 (2020).

[40] A. W. Cross, L. S. Bishop, J. A. Smolin, and J. M. Gambetta, Open quantum assembly language, arXiv:1707.03429 (2017).

[41] A. Kandala, K. Temme, A. D. Córcoles, A. Mezzacapo, J. M. Chow, and J. M. Gambetta, Error mitigation extends the computational reach of a noisy quantum processor, Nature **567**, 491 (2019).

[42] Y. Li and S. C. Benjamin, Efficient Variational Quantum Simulator Incorporating Active Error Minimization, Phys. Rev. X **7**, 021050 (2017).10.1103/PhysRevX.7.021050

[43] J. Chen and H. I. Nurdin, Correction to: Learning nonlinear input–output maps with dissipative quantum systems, Quantum Inf. Process. **18**, 354 (2019).

[44] D. Perez-Garcia, M. M. Wolf, D. Petz, and M. B. Ruskai, Contractivity of positive and trace-preserving maps under l_p norms, J. Math. Phys. **47**, 083506 (2006).

[45] S. Lang, *Complex Analysis*, Graduate Texts in Mathematics (Springer-Verlag, 1985).

[46] X. Ni, M. Verhaegen, A. J. Krijgsman, and H. B. Verbruggen, A new method for identification and control of nonlinear dynamic systems, Eng. Appl. Artif. Intell. **9**, 231 (1996).