Amplitude-based implementation of the unit step function on a quantum computer

Jonas Koppe * and Mark-Oliver Wolf

Department of Financial Mathematics, Fraunhofer Institute for Industrial Mathematics, 67663 Kaiserslautern, Germany

(Received 8 August 2022; accepted 19 January 2023; published 7 February 2023)

Modeling nonlinear activation functions on quantum computers is vital for quantum neurons employed in fully quantum neural networks, however, remains a challenging task. We introduce an amplitude-based implementation for approximating nonlinearity in the form of the unit step function on a quantum computer. Our approach expands upon repeat-until-success protocols, suggesting a modification that requires a single measurement only. We describe two distinct circuit types which receive their input either directly from a classical computer or as a quantum state when embedded in a more advanced quantum algorithm. All quantum circuits are theoretically evaluated using numerical simulation and executed on NOISY INTERMEDIATE-SCALE QUANTUM hardware. We demonstrate that reliable data with high precision can be obtained from our quantum circuits involving up to eight qubits and up to 25 CX-gate applications, enabled by state-of-the-art hardware-optimization techniques and measurement error mitigation.

DOI: 10.1103/PhysRevA.107.022606

I. INTRODUCTION

Identifying quantum algorithms that allow quantum speedups for machine learning is a research area of rising interest [1]. The integration of artificial neural networks with quantum computation is typically referred to as quantum neural networks [2–6]. A plethora of different construction strategies for quantum neural networks was reported over the last decade, a comprehensive review [7] of which, however, is far beyond the scope of this article. A rough categorization can be made between hybrid approaches that rely on variational quantum algorithms [8–12] and fully quantum implementations. For the latter, a key ingredient is the development of a quantum version for the Rosenblatt perceptron, involving the calculation of a tensor product, based on which subsequently a non-linear activation occurs [13]. While tensor calculus has already been performed on quantum computers [14–16], the implementation of nonlinear activation functions remains a challenging task [17–25].

In principle, nonlinearities can be modeled using the unit step function (USF), returning zero for negative function inputs and one otherwise. The USF is also known as the Heaviside (step) function or indicator function [26], originally developed in operational calculus, or the positive part of a function, commonly used in Fourier analysis [27] and finance [28]. Recently, different approaches to implement nonlinearity on a quantum computer were reported, involving phase encoding and inverse quantum Fourier transform [20], Taylor expansion [25], as well as bit-wise comparison [29–33]. Moreover, it has been known [17–19] that, specifically, the USF can in theory be implemented on quantum computers using repeat-until-success protocols [34–38]. However, a corresponding implementation has not been reported yet.

In this article, we suggest an approach that modifies the repeat-until-success gearbox circuit [34,37] to avoid midcircuit measurements, allowing to encode an arbitrarily close approximation of the USF on the amplitude of qubits. Specifically, we construct a quantum circuit performing a unitary transformation that, based on a given input, prepares a quantum register to represent the corresponding output of the USF. To this end, we structured the article as follows. First, we briefly outline the theoretical aspects of our implementation in Sec. II. We generally distinguish between passing the input to the circuit either directly as a floating-point value from a classical computer, referred to as *classical input*, or via another quantum state, referred to as *quantum-state input*. Therefore, we start with the simpler case of a classical input in Sec. III, where we first demonstrate the basic implementation for the USF, and subsequently augment the corresponding circuits for computing other nonlinear activation functions, e.g., the rectified linear unit (ReLU) [39]. Thereafter, in Sec. IV, we extend the discussion to the more complex scenario of a quantum-state input. There we first analyze input states representing a single angle only, and thereafter include in our analysis passing on quantum states representing multiple angles simultaneously, e.g., for computing the average value of the USF over a set of inputs.

The performance of all quantum circuits shown in this article is not only evaluated by numerical simulation assuming an ideal, noise-free quantum computer, but also tested on the IBM Quantum device in Ehningen, Germany. For implementing quantum circuits on Noisy Intermediate-Scale Quantum (NISQ) hardware [40], we found the number of employed CX gates to be the primary criterion for obtaining reliable results. Therefore, in the following we will restrict the characterization of quantum circuits on NISQ devices to the number of required CX-gate applications.

II. THEORETICAL CONSIDERATIONS

Encoding the output of the USF on the amplitude of a qubit can in principle be achieved by a transformation that takes as

*Corresponding author: jonas.koppe@ens-lyon.fr

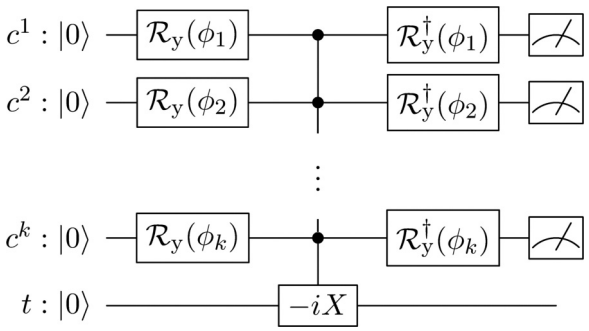

FIG. 1. A version of the gearbox circuit $C(\boldsymbol{\phi})$ suggested by Wiebe *et al.* [34,37].

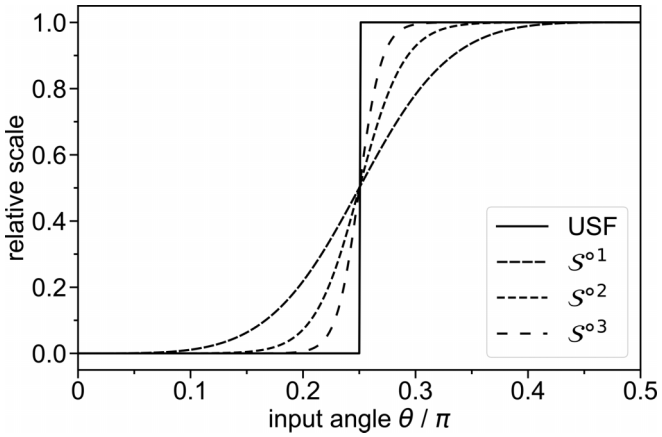

FIG. 2. USF and its approximations $\mathcal{S}^{\circ d}(\theta)$ [see Eq. (5)] for different levels $d = 1$, 2, and 3 on the input space used in this article.

an input an angle θ and performs a rotation on a qubit such that ideally

$$|0\rangle \longmapsto \begin{cases} |0\rangle & \text{for } 0 \leqslant \theta \leqslant \frac{\pi}{4}, \\ |1\rangle & \text{for } \frac{\pi}{4} < \theta \leqslant \frac{\pi}{2}. \end{cases} \tag{1}$$

The task of implementing the USF might thus be restated as using an input angle θ to create an effective rotation of the target qubit about 0 or π (see Fig. 2). We can approximate the transformation according to Eq. (1) with the help of the so-called gearbox circuit $C(\boldsymbol{\phi})$ demonstrated in Fig. 1 that was suggested by Wiebe *et al.* to synthesize an arbitrarily small rotation [34,37]. The name stems from the fact that the k input angles $\boldsymbol{\phi} = \phi_1, \ldots, \phi_k$ applied to the k control qubits $c^1, \ldots, c^k$ are used to produce a rotation involving a much finer rotation angle θ applied to the target qubit, where $\sin^2(\theta) = \sin^2(\phi_1) \cdots \sin^2(\phi_k)$. The entire transformation can formally be written as

$$\begin{aligned} C(\boldsymbol{\phi})\,|0^{\otimes k}\rangle_c\,|0\rangle_t = \rho(\theta)\,|0^{\otimes k}\rangle_c\; e^{-i\arctan\left[\tan^2\theta\right]X}\,|0\rangle_t \\ - \sqrt{1-\rho^2(\theta)}\,|\emptyset^{\otimes k}\rangle_c\; e^{-i\frac{\pi}{4}X}\,|0\rangle_t\,, \end{aligned} \tag{2}$$

where we used the indices c and t to indicate the control (qubit) register and target qubit, respectively. $|\emptyset^{\otimes k}\rangle_c$ is a compact notation representing the sum over all possible states of the control register except for the all-zero state $|0^{\otimes k}\rangle_c$. Notably, a rotation involving θ, i.e., the transformation $|0\rangle_t \mapsto e^{-i\arctan[\tan^2\theta]X}\,|0\rangle_t$, is successfully applied to the target qubit only if the control register is in $|0^{\otimes k}\rangle_c$. The corresponding success probability is given by

$$\rho^2(\theta) := \sin^4(\theta) + \cos^4(\theta). \tag{3}$$

Otherwise $C(\boldsymbol{\phi})$ transforms the target qubit according to $|0\rangle_t \mapsto e^{-i\frac{\pi}{4}X}\,|0\rangle_t$. Without exactly determining the state of the target qubit after circuit execution, measuring the control register consequently allows to ascertain which transformation was performed on the target qubit. Repeating the circuit $C(\boldsymbol{\phi})$ until the state $|0^{\otimes k}\rangle_c$ is measured in turn ensures that the desired transformation $|0\rangle_t \mapsto e^{-i\arctan[\tan^2\theta]X}\,|0\rangle_t$ has occurred. This is known as the repeat-until-success principle [17,35–38]. Moreover, Wiebe *et al.* theoretically showed that a nested version of the gearbox circuit can be built up recursively, i.e., the transformation on the target qubit from the gearbox circuit of Fig. 1 serves itself as the input of an outer gearbox circuit as shown in Fig. 3(b). The transformation performed on the outermost target qubit of a d-times nested gearbox circuit given all control qubits are found in $|0\rangle_c$ is then given as $|0\rangle_t \mapsto e^{-i\arctan[\tan^{2^d}\theta]X}\,|0\rangle_t$ [34,37]. It is emphasized that generally the success probability $\rho^2(\theta)$ depends on d, as detailed in the Supplemental Material (Sec. SI 1) [41]. However, in the following only the case for $d = 1$ as given in Eq. (3) will be relevant.

For implementing an approximation of the USF we use the square-wave property of the gearbox [37]. Therefore, the most basic gearbox version is sufficient, involving only a single control qubit, and likewise a single input angle $\phi_1 = \theta$, as shown in Fig. 3(a). It is emphasized that accordingly the entire complex circuit shown in Fig. 1 reduces to a very basic circuit, which is shown in Fig. 3(e) without measurements [18], in the following referred to as a single-step gearbox $\mathcal{G}b(\theta)$. The transformation from Eq. (2) can then be expressed as

$$\begin{aligned} \mathcal{G}b(\theta)\,|0\rangle_c\,|0\rangle_t = \rho(\theta)\,|0\rangle_c\; \big(\cos[\arctan[\tan^2\theta]]\,|0\rangle_t \\ + \sin[\arctan[\tan^2\theta]]\,|1\rangle_t\big) \\ - \sin(\theta)\cos(\theta)\,|1\rangle_c\; (|0\rangle_t + |1\rangle_t). \end{aligned} \tag{4}$$

Assuming the control qubit to be in the zero state $|0\rangle_c$, the probability of finding the target qubit in $|1\rangle_t$ is given by $\mathcal{S}^{\circ 1}(\theta) := \sin^2[\arctan[\tan^2\theta]]$; the corresponding trajectory is demonstrated in Fig. 2. Clearly $\mathcal{S}^{\circ 1}(\theta)$ is reminiscent of a sigmoid function [42] for $\theta \in [0, \pi/2]$, and thus an appropriate approximation for the USF. Extending this observation to a d-nested gearbox circuit, in the following referred to as a d-step gearbox, the probability of finding the target qubit in $|1\rangle_t$, provided all control qubits were previously found in $|0\rangle_c$, is accordingly given by

$$\mathcal{S}^{\circ d}(\theta) := \sin^2[\arctan[\tan^{2^d}\theta]]. \tag{5}$$

Trajectories for $d = 2$ and 3 are likewise shown in Fig. 2. Even though it is theoretically known that in such a way an approximation of the USF is encoded in the amplitude of the target qubit [17,37], to the best of our knowledge, an exact protocol for exploiting this fact has not been reported yet. Over the course of this article we will introduce two

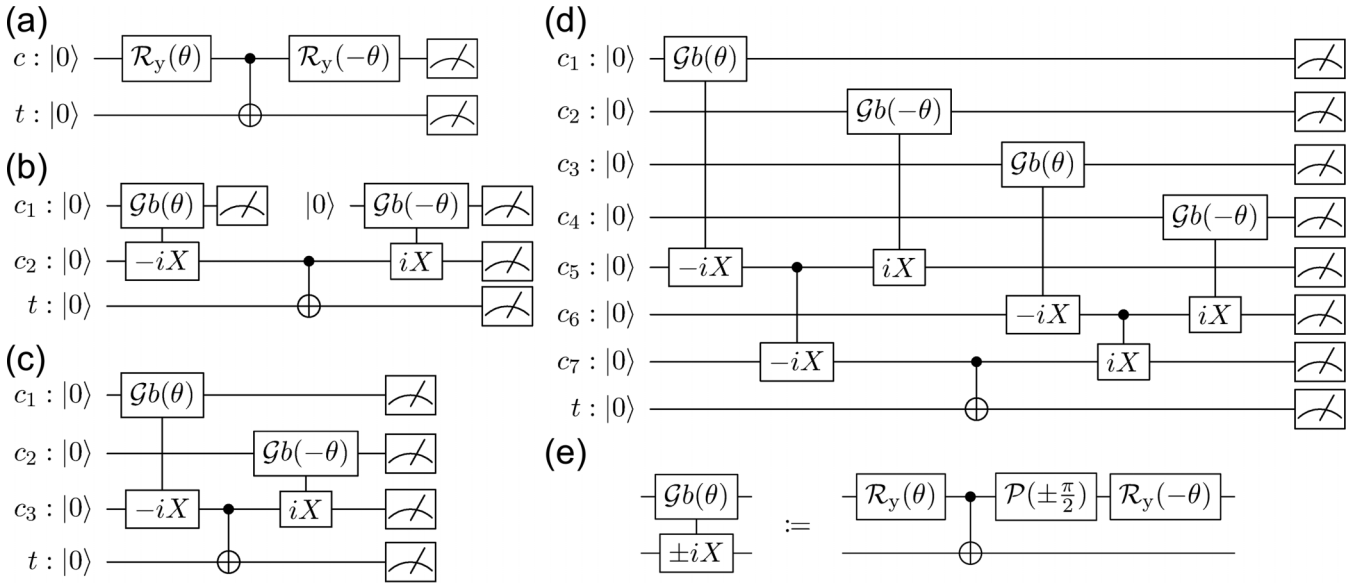

FIG. 3. Quantum circuits implementing the USF for different levels of approximation d. (a) Single-step gearbox with $d = 1$, (b) double-step gearbox with $d = 2$ according to repeat-until-success protocols [34], (c) double-step gearbox with $d = 2$, and (d) triple-step gearbox with $d = 3$ as suggested in this article. (e) Definition of the elementary gearbox-circuit element used in (b), (c), and (d), which does not comprise any measurements. Numbers of the control qubits are given as subscripts to distinguish them from the control qubits shown in Fig. 1.

alternatives for retrieving $\mathcal{S}^{\circ d}(\theta)$ from the quantum states produced by the d-step gearbox circuits. The application of both is demonstrated for the two scenarios described above, i.e., the classical input, where the input for the gearbox circuit is a single angle $\theta \in [0, \pi/2]$ directly passed on by a classical computer, based on which the target qubit is ideally transformed according to Eq. (1) as demonstrated in Fig. 2 (USF). Then the task is extended to passing a quantum-state input to the gearbox circuit, where each eigenstate in the measurement basis represents a different input angle $\theta_j \in [0, \pi/2]$ used for the transformation in Eq. (1).

III. CLASSICAL INPUT

A. Basic USF version

Consider the single-step gearbox circuit from Fig. 3(a), where we wish the measurement to reflect $\mathcal{S}^{\circ 1}(\theta)$. Evaluation of Eq. (4), however, shows that measuring the probability for the state $|1\rangle_t |0\rangle_c$ yields $\rho^2(\theta)\mathcal{S}^{\circ 1}(\theta)$. It is indeed possible to remove the distortion $\rho^2(\theta)$ by combining the probabilities measured for both states involving $|0\rangle_c$: the probability of measuring the target qubit in state $|1\rangle_t$ under the condition that the control qubit is found in state $|0\rangle_c$ is given by

$$\begin{aligned}\Omega(1 \mid 0) &:= \mathrm{P}(|1\rangle_t \mid |0\rangle_c) = \frac{|(\langle 1|_t \langle 0|_c)\,|\tau_{\mathcal{G}b}\rangle|^2}{|(\langle 0|_t \langle 0|_c)\,|\tau_{\mathcal{G}b}\rangle|^2 + |(\langle 1|_t \langle 0|_c)\,|\tau_{\mathcal{G}b}\rangle|^2} \\ &= \frac{\rho^2(\theta)\sin^2[\arctan[\tan^2\theta]]}{\rho^2(\theta)\cos^2[\arctan[\tan^2\theta]] + \rho^2(\theta)\sin^2[\arctan[\tan^2\theta]]} = \sin^2[\arctan[\tan^2\theta]] = \mathcal{S}^{\circ 1}(\theta), \end{aligned} \tag{6}$$

where $|\tau_{\mathcal{G}b}\rangle$ represents the state of both gearbox qubits before initiating the measurement. Unfortunately, this evaluation cannot simply be extended to d-step gearbox circuits and thus to higher levels of approximation d for the USF, as they involve repeat-until-success protocols that rely on mid-circuit measurements [17,34–38] [cf. Fig. 3(b)]. We propose an alternative implementation by allocating each gearbox element within a nested circuit to a new set of qubits, as demonstrated in the circuit modification in Fig. 3(c) for the double-step gearbox. In Fig. 3(d), the corresponding extension of the triple-step gearbox is shown. We emphasize that our implementation, combined with appropriate data evaluation as given in Eq. (6), allows us to postpone all measurements to the end of the circuit and with this to omit any intermediate interaction with the involved qubits. Our approach scales with the level of approximation d, which principally requires 2^d measurable qubits, and the application of $2^d - 1$ CX gates. The corresponding generalization of Eq. (6) is given by

$$\Omega(1 \mid 0^{\otimes 2^d-1}) = \sin^2[\arctan[\tan^{2^d}\theta]] = \mathcal{S}^{\circ d}(\theta). \tag{7}$$

For a more thorough mathematical treatment, the reader is referred to the Supplemental Material (Sec. SI 1) [41]. To demonstrate the validity of this approach, we performed simulations of the quantum circuits shown in Figs. 3(c) and 3(d), implementing the approximation level $d = 2$ (double-step gearbox) and $d = 3$ (triple-step gearbox) for the unit step function, respectively. An ideal, noise-free quantum computer with all-to-all connectivity was assumed, requiring four qubits and three CX gates for $d = 2$, and eight qubits and

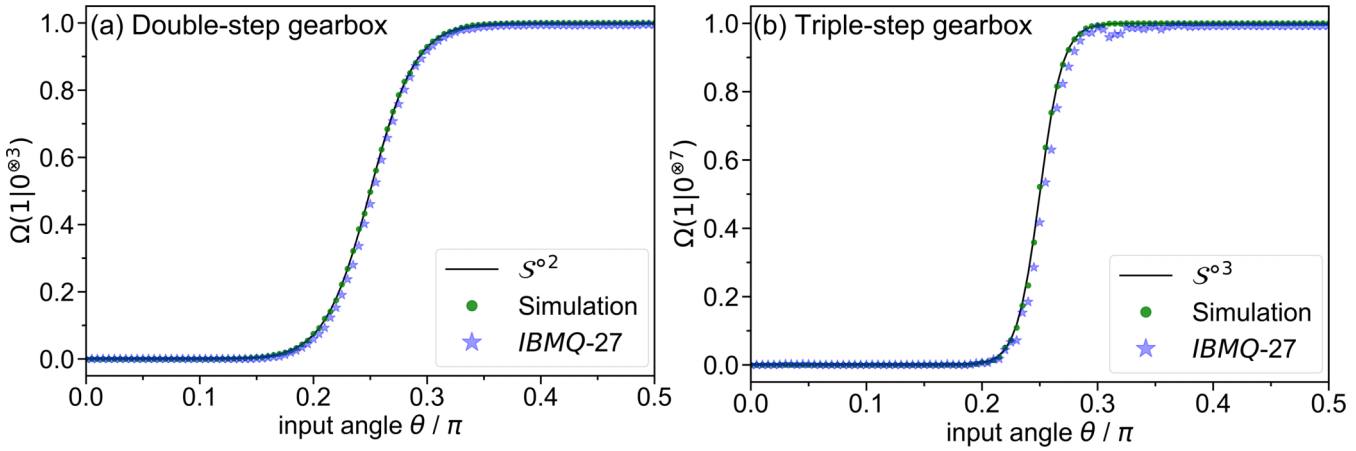

FIG. 4. Results for the implementation of the USF at approximation levels $d = 2$ and 3 in (a) and (b), respectively. Analytical curves are based on Eq. (5). Simulated data (green circles) were obtained by executing the quantum circuits shown in Figs. 3(c) and 3(d). For the data obtained from *IBMQ-27 Ehningen* (blue stars), the respective circuits were optimized prior to execution. Optimum quantum circuits are demonstrated in Fig. S3. For all circuit runs, the input space $\theta \in [0, \pi/2]$ was covered by 101 increments $\theta_j = j\pi/200$, $j = 0, \ldots, 100$, where for each data point 10^5 circuit executions were conducted.

seven CX gates for $d = 3$. The input space $\theta \in [0, \pi/2]$ was covered by 101 increments, where each data point represents an estimate for $\Omega(1 \mid 0^{\otimes 3})$ and $\Omega(1 \mid 0^{\otimes 7})$, obtained from 10^5 circuit executions. The results (green circles) are compared to $\mathcal{S}^{\circ 2}(\theta)$ and $\mathcal{S}^{\circ 3}(\theta)$ in Fig. 4. Clearly, the numerical simulations are in excellent agreement with their respective analytical form, confirming Eq. (7). To assess the applicability of the implementation on NISQ devices, both quantum circuits were optimized for the 27-qubit IBM Quantum system in Ehningen, Germany (*IBMQ-27 Ehningen*). The hardware-optimization procedure is detailed in Appendix B, the characteristics of *IBMQ-27 Ehningen* are summarized in Appendix C. It should be emphasized here that, due to the restricted connectivity based on the heavy-hexagon qubit architecture, state-swapping was required for the triple-step gearbox circuit, resulting in an overhead of four CX gates, and thus, in 11 CX-gate applications in total. The results are likewise shown in Fig. 4 (blue stars). Again, 10^5 circuit executions were performed for each input angle θ. In general, the data reflects the simulated behavior with high precision. Yet deviations do occur for input angles at about the step $\theta \approx \pi/4$, where the values obtained from *IBMQ-27 Ehningen* systematically remain below the numerical prediction. Even though this effect becomes more pronounced for the triple-step gearbox, it does not qualitatively affect the approximation of the USF.

B. Variations

We emphasize that the evaluation strategy described above for retrieving $\mathcal{S}^{\circ d}(\theta)$ can be incorporated into more advanced quantum circuits. For instance, consider a unitary $\mathcal{U}$ requiring as an input the (approximated) USF $\mathcal{S}^{\circ d}(\theta)$, which performs a transformation on an additional input state $|\Phi\rangle$, and maps the result to an output qubit o. A schematic representation of this type of circuit involving the double-step gearbox is suggested in Fig. 5(a). It is important to highlight that only the state of the target qubit t must be passed to $\mathcal{U}$. Eventually, data evaluation is done analogously to Eq. (7), however, replacing the target qubit t with the output qubit o, i.e.,

$$\tilde{\Omega}(1 \mid 0^{\otimes 2^d-1}) := \mathrm{P}(|1\rangle_o \mid |0^{\otimes 2^d-1}\rangle_c). \tag{8}$$

As a primitive example for $\mathcal{U}$ requiring no additional input $|\Phi\rangle$, assume we wish to modify the transformation given in Eq. (1) and instead encode

$$|0\rangle_o \longmapsto \begin{cases} \cos\kappa\,|0\rangle_o + \sin\kappa\,|1\rangle_o & \text{for } 0 \leqslant \theta \leqslant \frac{\pi}{4}, \\ |1\rangle_o & \text{for } \frac{\pi}{4} < \theta \leqslant \frac{\pi}{2}, \end{cases} \tag{9}$$

on the output qubit o. This effectively allows shifting the first plateau from 0 to $\sin^2\kappa$. In Fig. 5(b) we demonstrate a quantum circuit approximating this transformation using a double-step gearbox, such that we expect to observe the function $\tilde{\mathcal{S}}^{\circ 2}(\theta) := \sin^2(\theta)\kappa(1 - \mathcal{S}^{\circ 2}(\theta)) + \mathcal{S}^{\circ 2}(\theta)$. Another important transformation that can be generated from the USF is the so-called ReLU activation function [39], commonly defined as $\mathrm{ReLU}(x) := \max(0, x)$. Here, the output of the USF is multiplied with the identity $f(x) = x$. Note that the unit step must necessarily coincide with the root of $f(x)$. For the implementation on a quantum computer this requires synchronization of the gearbox input θ and the function input x. A variety of strategies could be used for this purpose. For simplicity, here we rigorously encode $f(x)$ on a single input qubit q_1 using an $\mathcal{R}_y(x)$ gate with

$$x = x(\theta) = \arcsin\left(\sqrt{\left|\frac{2\theta}{\pi} - \frac{1}{2}\right|}\right), \tag{10}$$

where we make use of the fact that for all $x \leqslant \pi/4$ the corresponding gearbox output $\mathcal{S}^{\circ d}(\theta)$ evaluates to zero. We thereby guarantee that $f[x(\theta)] \in [0, 1/2]$ and $f[x(\pi/4)] = 0$. The entire ReLU transformation, formally written as

$$|0\rangle_o \longmapsto \begin{cases} |0\rangle_o & \text{for } 0 \leqslant \theta \leqslant \frac{\pi}{4}, \\ \sqrt{f(x(\theta))}\,|1\rangle_o & \text{for } \frac{\pi}{4} < \theta \leqslant \frac{\pi}{2}, \end{cases} \tag{11}$$

is approximated by the quantum circuit demonstrated in Fig. 5(d), employing a double-step gearbox. The expected

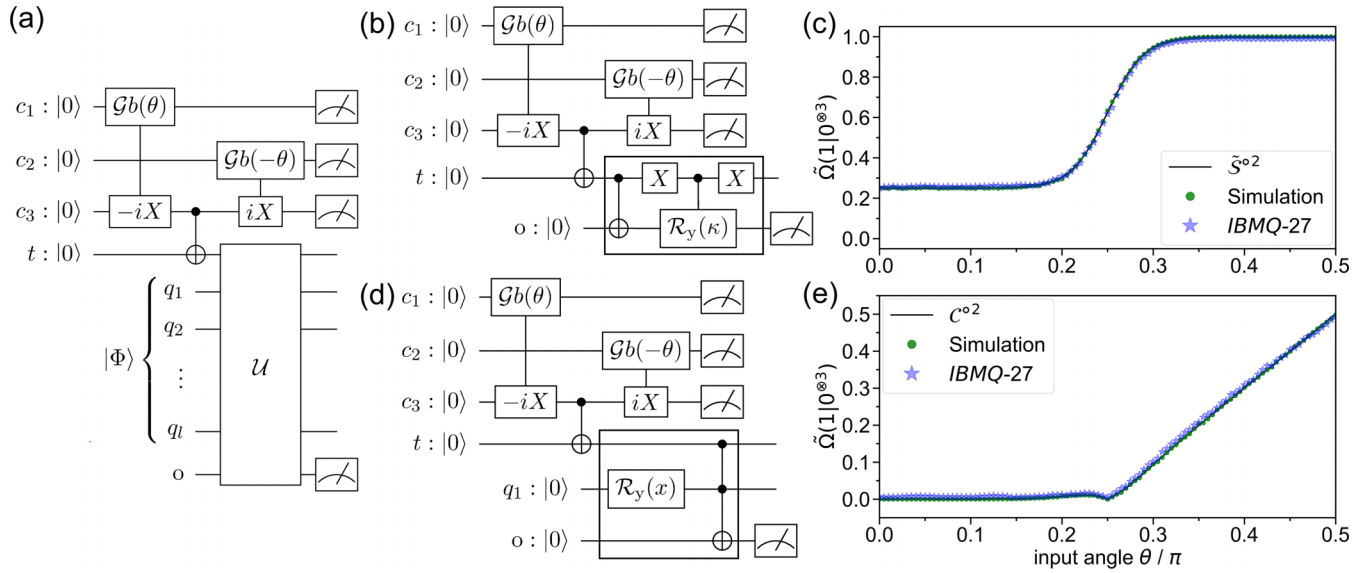

FIG. 5. (a) Schematic representation for combining an arbitrary unitary transformation $\mathcal{U}$ with the double-step gearbox from Fig. 3(c). (b), (d) Quantum circuits approximating the transformations given in Eqs. (9) and (11), respectively. The black boxes indicate the additional unitary transformations $\mathcal{U}$ as generally introduced in (a). (c), (e) Data obtained from executing the quantum circuits shown in (b), (d) are compared to the respective analytical forms. All conditions are identical to those reported for Fig. 4. Optimum quantum circuits on *IBMQ-27 Ehningen* are shown in Figs. S1 and S2.

output is then given by $C^{\circ 2}(\theta) \coloneqq \mathcal{S}^{\circ 2}(\theta)\, f[x(\theta)]$. Using identical conditions as described in Fig. 4, we tested the performance of the quantum circuit from Fig. 5(b) for $\kappa = \arcsin\sqrt{1/4}$, i.e., expecting a first plateau at $1/4$, and the quantum circuit from Fig. 5(d) on an ideal, noise-free quantum computer as well as on *IBMQ-27 Ehningen*. The corresponding data are compared to the analytical forms in Figs. 5(c) and 5(e), respectively, and again found in excellent agreement.

IV. QUANTUM-STATE INPUT

The results discussed in the previous section generally demonstrate the ability to approximate the USF on a quantum computer. However, directly passing an input angle θ to the gearbox circuit requires communication with a classical computer. For many future applications, and in particular for designing a fully quantum neural network, we rather assume gearbox circuits to receive a quantum state $|\Psi\rangle$ as an input, with each eigenstate of the measurement basis $|\psi_j\rangle$ representing a different input angle θ_j. In this section, we therefore demonstrate results for a single-step gearbox capable of considering four arbitrarily chosen input angles $\theta/\pi \coloneqq (0.15, 0.2, 0.4, 0.45)^T$, where each entry is denoted as θ_j/π, $j = 0, \ldots, 3$. These angles are represented by a two-qubit state register s, i.e., by the amplitudes of $|\psi_j\rangle \in \{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, respectively. An efficient approach for passing input angles to the gearbox circuit based on the state of s are uniformly controlled rotations [43,44]. This generally requires 2^p CX-gate applications, where p indicates the number of qubits in the state register. The corresponding circuit element for a two-qubit state register is shown in Fig. 6(a). Note that the angles ϑ_j involved in the depicted sequence of rotations can be obtained from the θ_j, the exact conversion is detailed in [43]. The incorporation of uniformly controlled rotations into the single-step gearbox, as demonstrated in Fig. 6(b), generally requires nine CX gates.

A. Single angle

At first we consider a situation where the state-qubit register is in one of the eigenstates of the measurement basis, i.e., $|\Psi\rangle = |\psi_j\rangle$, such that only one of the four angles θ_j is passed to the gearbox circuit. In fact, this is not fundamentally different from the case of a classical input discussed in the previous section; data evaluation can be performed according to Eq. (6) to obtain $\mathcal{S}^{\circ 1}(\theta_j)$. In Fig. 7, $\mathcal{S}^{\circ 1}(\theta)$ is again shown for the entire input space $\theta \in [0, \pi/2]$. The results

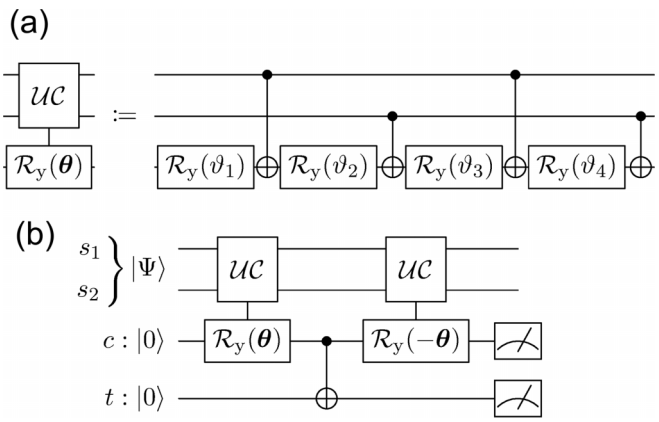

FIG. 6. (a) Uniformly controlled rotations involving two state qubits. For eventually implementing the correct rotation according to $\theta_j/\pi \in \{0.15, 0.2, 0.4, 0.45\}$, angles must be converted as described in [43]. The corresponding angles here are given by $\vartheta_j/\pi \in \{0.3, -0.025, 0, -0.125\}$. (b) Uniformly controlled rotations as shown in (a) embedded in the single-step gearbox. The state register is represented by $|\Psi\rangle = \sum_j \alpha_j |\psi_j\rangle$.

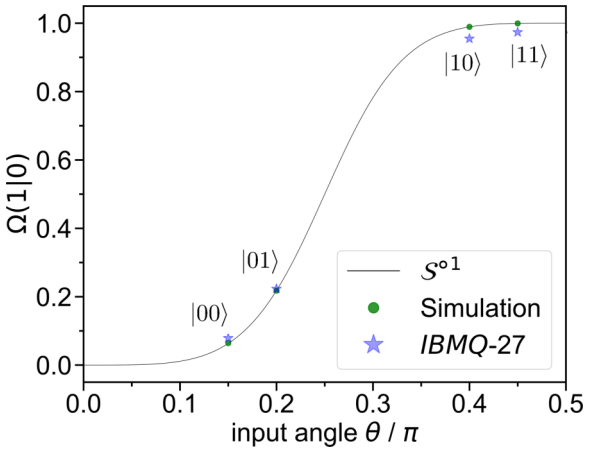

FIG. 7. Results for the quantum-state input passed to the single-step gearbox as shown in Fig. 6(b). The state register is represented by $|\Psi\rangle \in \{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$ as indicated, corresponding to the input angles $\theta_j/\pi \in \{0.15, 0.2, 0.4, 0.45\}$. The curve for $\mathcal{S}^{\circ 1}(\theta)$ is based on Eq. (5). Simulated data, illustrated as green circles, stem from executing the quantum circuits shown in Fig. 6(b) 10^5 times assuming an ideal, noise-free quantum computer with all-to-all connectivity. For data obtained from *IBMQ-27 Ehningen*, the circuits were again optimized. One hundred hardware runs were performed, each comprising 10^5 circuit executions. The blue stars represent the respective averages. The optimum quantum circuit for $|00\rangle$ on *IBMQ-27 Ehningen* is shown in Fig. S4(a).

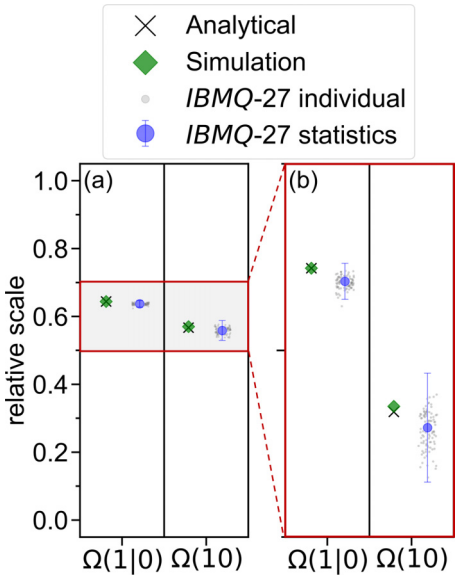

FIG. 8. (a) Results for the average output of the single-step gearbox from Fig. 6(b). (b) Inset indicated with the red box in (a). Analytical values are given by $\Omega(1 \mid 0) \approx 0.644$ according to Eq. (12) and $\overline{\mathcal{S}^{\circ 1}}(\theta) = 0.567$ for $\Omega(10)$. Data for $\Omega(1 \mid 0)$ are based on executing the circuit from Fig. 6(b) with the state register represented by $|\Psi\rangle = |+\rangle\,|+\rangle$. Data for $\Omega(10)$ stems from executing the subcircuits Δ^0, Δ^2, Δ^4, and Δ^6, and postprocessing the results as detailed in the main text. For all (sub)circuits, 100 hardware runs (gray circles) with 10^5 executions each were conducted. The respective averages and 3σ levels are indicated by the blue circles and error bars. The optimum quantum circuits on *IBMQ-27 Ehningen* for $\Omega(1 \mid 0)$ and $\Omega(10)$ can be found in Fig. S4.

for executing the quantum circuit from Fig. 6(b) 10^5 times for each $|\Psi\rangle = |\psi_j\rangle$ assuming an ideal, noise-free quantum computer are represented by the green circles. Additionally, the respective states of the state register are indicated. Clearly, the analytical and simulated data are in excellent agreement. For assessing its performance on a NISQ device, the quantum circuit from Fig. 6(b) was again optimized for *IBMQ-27 Ehningen* (resulting in nine CX-gate applications with no overhead), and executed 10^5 times for each input state. Each of these runs was then repeated 100 times, from which the resulting average values for $\Omega(1 \mid 0)$ are indicated by the blue stars in Fig. 7. Notably, the data obtained from *IBMQ-27 Ehningen* do not achieve the level of precision we expect from the results demonstrated in Fig. 4: while for $\theta_j < \pi/4$ estimators for $\Omega(1 \mid 0)$ are found slightly above the expected values, for $\theta_j > \pi/4$ results remain below the analytical and numerical prediction. We assign this to the diminished precision of θ_j passed to the gearbox. Here θ_j is generated by the uniformly controlled rotations, involving four CX-gates and several single-qubit-gate applications, and is thus significantly more prone to error when compared to the classical input θ from the previous section. Since the individual data obtained from *IBMQ-27 Ehningen* (100 hardware runs not shown here for clarity) have sufficiently small confidence intervals (3σ levels lie within the size of the symbol), this error seems to be rather systematic than random. Nevertheless, the overall behavior according to $\mathcal{S}^{\circ 1}(\theta)$ is generally well reflected by these results.

B. Multiple angles

A more complex situation occurs when the state register s is no longer in one of the four eigenstates $|\psi_j\rangle$. Here we are particularly concerned with the state qubits in an equal superposition of all four eigenstates $|\Psi\rangle = |+\rangle\,|+\rangle = 1/2 \sum_{j=0}^{3} |\psi_j\rangle$, i.e., the average output of the gearbox. Unfortunately, the success probability $\rho^2(\theta_j)$ is state dependent, such that $\overline{\mathcal{S}^{\circ 1}}(\theta) := 1/4 \sum_{j=0}^{3} \mathcal{S}^{\circ 1}(\theta_j)$ cannot simply be retrieved by using the evaluation strategy suggested in Eq. (6). Instead, this yields

$$\Omega(1 \mid 0) = \frac{\sum_{j=0}^{3} \rho^2(\theta_j) \sin^2[\arctan[\tan^2 \theta_j]]}{\sum_{j=0}^{3} \rho^2(\theta_j)}, \tag{12}$$

where j is the integer representation of the two-bit string in the computational basis. For the chosen input angles θ, Eq. (12) evaluates to $\Omega(1 \mid 0) \approx 0.644$, while we wish to find $\overline{\mathcal{S}^{\circ 1}}(\theta) \approx 0.567$. To confirm this bias, we repeated the procedure described for Fig. 7 with the state register given by $|\Psi\rangle = |+\rangle\,|+\rangle$. All data are demonstrated in the left panel of Fig. 8(a). Here, we additionally showed the individual 100 hardware runs in *IBMQ-27 Ehningen* as small gray circles. Blue circles and error bars indicate the average and 3σ levels, respectively. The inset indicated by the red box is shown in Fig. 8(b). For clarity, analytical and simulated data are shifted horizontally with respect to the data obtained from *IBMQ-27 Ehningen*. Generally, these results are in agreement with Eq. (12). The estimator for $\Omega(1 \mid 0)$ based on data obtained from *IBMQ-27 Ehningen* remains slightly below the predicted

value, which, however, is in accordance with the observations from Fig. 7.

It should be reiterated here that, following Eq. (6), for the single-step gearbox the probability of measuring $|0\rangle_c\,|1\rangle_t$ with a state register in equal superposition is given by

$$|(\langle 1|_t\,\langle 0|_c)\,|\tau_{\mathcal{G}b}\rangle|^2 = \frac{1}{4}\sum_{j=0}^{3}\rho^2(\theta_j)\sin^2[\arctan[\tan^2\theta_j]], \quad (13)$$

with $|\tau_{\mathcal{G}b}\rangle$ representing the state for the two involved gearbox qubits. Aiming to obtain $\overline{\mathcal{S}^{\circ 1}}(\theta)$ from this result, a transformation must be found performing a state-wise amplitude encoding for $\rho^{-1}(\theta_j)$, which is then appended to the gearbox circuit. Then, a measurement yields

$$\Omega(10) \coloneqq \frac{1}{4}\sum_{j=0}^{3} D(\theta_j)\,\rho^2(\theta_j)\sin^2[\arctan[\tan^2\theta_j]] = \overline{\mathcal{S}^{\circ 1}}(\theta), \quad (14)$$

where $D(\theta_j) \coloneqq \rho^{-2}(\theta_j)$. Since $\rho^{-1}(\theta_j)$ is a periodic function [cf. Eq. (3)], a corresponding transformation may be approximated using its Fourier series, a strategy which was suggested for similar purposes before [45]. However, due to the structure of the problem, here we chose a different approach than previously reported. Each term from the series expansion is encoded on a different qubit. Eventually, the full transformation can be constructed by adding up all of these terms using amplitude addition [31]. Note that this approach requires an approximation of $D(\theta_j)$ instead of $\rho^{-1}(\theta_j)$ and likewise the consideration of probabilities instead of amplitudes in the Fourier series expansion, where the latter can be accounted for by employing squared sinusoidal functions. Using simple trigonometric identities, the first four terms of the corresponding approximation are given by

$$\frac{D(\theta_j)}{2} \approx 0.915 - 0.485\cos^2(2\theta_j) + 0.083\cos^2(4\theta_j) - 0.014\cos^2(6\theta_j). \quad (15)$$

The factor $1/2$ must be included to ensure normalization. The error that results from the approximation given in Eq. (15) is significantly smaller than the error due to current NISQ hardware. A more detailed description is provided in the Supplemental Material (Sec. SI 6). In principle, it is possible to construct a quantum circuit that entirely computes Eq. (15) on the amplitude of a single qubit and multiply it to the gearbox output according to Eq. (14), i.e.,

$$\frac{\Omega(10)}{2} \approx \underbrace{\frac{1}{4}\sum_{j=0}^{3} D(\theta_j)\,\rho^2(\theta_j)\sin^2[\arctan[\tan^2\theta_j]]}_{\text{circuit }\Delta}. \quad (16)$$

After multiplying the result of the measurement for Δ with the factor of 2, indeed $\overline{\mathcal{S}^{\circ 1}}(\theta)$ is obtained. The construction of the quantum circuit Δ including amplitude subtraction is detailed in the Supplemental Material (see Sec. SI 4 and following). Before proceeding, we note that the magnitude of the bias, i.e., the deviation between $\Omega(1\mid 0)$ and $\overline{\mathcal{S}^{\circ 1}}(\theta)$ depends on the set of chosen angles θ, which, however, does not affect the performance of the outlined procedure.

The implementation of Δ theoretically requires 238 CX-gate applications and is far beyond what we expect to be reasonably implementable on state-of-the-art NISQ hardware. Therefore, in the remaining part of this section we present a strategy for obtaining reliable results for Δ on *IBMQ-27 Ehningen* regardless. The decisive advantage from our approach for implementing the Fourier sSeries expansion of $D(\theta_j)$ can be demonstrated by rewriting Eq. (16) as

$$\frac{\Omega(10)}{2} \approx \sum_{k=0}^{3}(-1)^k \underbrace{\frac{1}{4}\sum_{j=0}^{3} D^{2k}(\theta_j)\,\rho^2(\theta_j)\sin^2[\arctan[\tan^2\theta_j]]}_{\text{subcircuit }\Delta^{2k}}, \quad (17)$$

where $D^{2k}(\theta_j) = a_{2k}\cos^2(2k\theta_j)$ and a_{2k} stemming from Eq. (15). Accordingly, instead of performing the full transformation by a single, complex circuit Δ, we are able to split Δ into four subcircuits Δ^{2k}, implementing the $D^{2k}(\theta_j)$ elements separately. After execution, the individual results are then postprocessed as $\Delta = \Delta^0 - \Delta^2 + \Delta^4 - \Delta^6$ to eventually obtain $\overline{\mathcal{S}^{\circ 1}}(\theta)$. Each subcircuit Δ^{2k} begins with the single-step gearbox as shown in Fig. 6(b) with the state register represented by $|\Psi\rangle = |+\rangle\,|+\rangle$. Since subcircuit Δ^0 constitutes a specifically simple case, which only includes rescaling the gearbox result by a_0 [cf. Eqs. (15) and (17)], its discussion is postponed for the moment. The Δ^λ, $\lambda = 2, 4, 6$ subcircuit family is more elaborate: an additional uniformly controlled rotation is required to encode the respective term $\cos^2(\lambda\theta_j)$, which is subsequently rescaled by a_λ, and finally multiplied to the gearbox result. The corresponding quantum circuit is demonstrated in the top row of Fig. 9(a). In the lower row, the corresponding topological requirement for a quantum computer is schematically shown using color codes and lines for indicating direct communication between the involved qubits. Assuming all-to-all connectivity, a Δ^λ subcircuit requires 25 CX gates. Considering the typical heavy-hexagon architecture of *IBMQ* devices, even the most efficient hardware realization found for Δ^λ involves 44 CX-gate applications and thus still remains too technically demanding. The last resource available for further simplifying the Δ^{2k} subcircuits remains in externalizing the multiplication by a_λ to a classical computer. The corresponding quantum circuit, shown in the upper panel of Fig. 9(b), solely rescales the gearbox result with the respective $\cos^2(\lambda\theta_j)$ term and stores the value in a qubit $t\Delta^\lambda$. The final result is then obtained postmeasurement by classically computing $\Delta^\lambda = t\Delta^\lambda\,a_\lambda$. This modification in turn allows further simplifying the last multiplication step implemented by the Toffoli gate $\mathcal{T}$ [46] (green-colored box in Fig. 9): even the most efficient implementation requires 6 CX-gate applications and all-to-all connectivity of the three involved qubits [47,48]. However, a phase-equivalent version $\tilde{\mathcal{T}}$ was reported, performing the identical transformation as $\mathcal{T}$, but additionally flagging the $|101\rangle$ state (referring to the three involved qubits) with a negative amplitude (phase shift by π) [49,50]. This version can be implemented using only three CX gates, and additionally does not require communication between the two control qubits. Both hardware implementations for the Toffoli gate and the phase equivalent version can be found in the Supplemental Material (Fig. S6). Since $\mathcal{T}$ in Fig. 9(b) is immediately followed by a measurement, $\tilde{\mathcal{T}}$ can

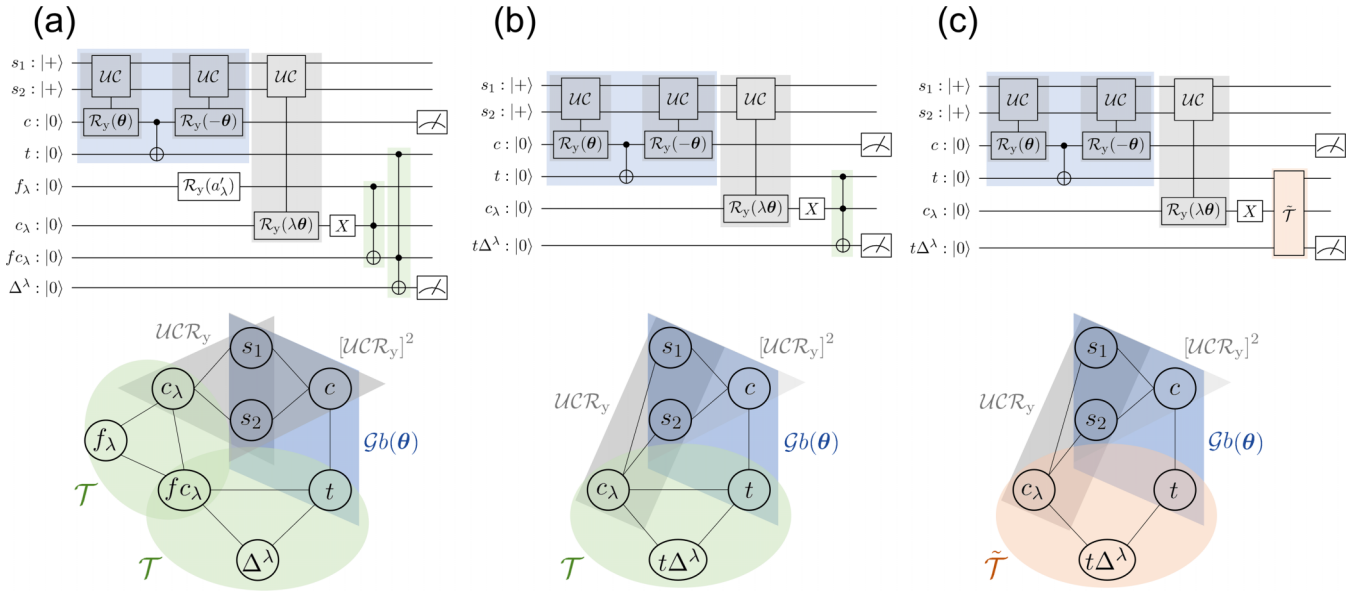

FIG. 9. The upper row shows quantum circuits for different variants of the Δ^λ subcircuit family. (a) Full subcircuit, (b) subcircuit with externalized multiplication by a_λ, and (c) as in (b) but with the phase-equivalent version of the Toffoli gate. Generally, following the gearbox procedure, all values are encoded in the amplitude for the $|1\rangle$ state of each qubit, i.e., qubits f_λ, c_λ, and fc_λ store the values for the coefficient $\sqrt{a_\lambda}$, the cosine term $\cos(\lambda\theta_j)$, and their product $\sqrt{a_\lambda}\cos(\lambda\theta_j)$, respectively. Qubits Δ^λ and $t\Delta^\lambda$ represent final circuit results, where the second term indicates that a further postprocessing step is required. Coefficients $\sqrt{a_\lambda}$ are encoded using a $\mathcal{R}_y(a'_\lambda)$ gate with $a'_\lambda = \arcsin(\sqrt{a_\lambda})$. Topological maps in the lower row demonstrate the respectively required connectivity on a quantum computer. Lines indicate direct communication between the involved qubits. Subcircuit elements are identified using color coding. The number of CX gates are summarized in Table I.

be employed regardless. This is demonstrated in Figure 9(c), and ultimately allowed us to reduce the number of CX gates to 16 assuming all-to-all connectivity, and to 25 considering the architecture of *IBMQ-27 Ehningen*. All requirements for the quantum circuits from Fig. 9 are summarized in Table I.

Note that multiplying a_λ postmeasurement can likewise be employed for Δ^0, leaving the circuit at the same complexity as described in Fig. 7 (nine CX-gate applications, no overhead). In Fig. 8(a), right panel, the analytical value $\overline{\mathcal{S}^{\circ 1}}(\theta) \approx 0.567$ is compared to the estimators for $\Omega(10)$ according to Eq. (14). Data for $\Omega(10)$ were obtained by performing runs of the four subcircuits Δ^λ, $\lambda = 0, 2, 4, 6$ in their most simplified version on an ideal, noise-free quantum computer assuming all-to-all connectivity, and on the *IBMQ-27 Ehningen* device using the optimum hardware realizations. The inset indicated by the red box is again given in Fig. 8(b). The overall procedure is identical to the one reported for $\Omega(1 \mid 0)$, shown in the left panel. Notably, the simulated result for $\Omega(10)$ (green diamond) is slightly larger than the analytical value (black cross). This is in accordance with stopping the series expansion of $D(\theta_j)$ after a negative term, leaving the approximation below the exact value. Regarding the 100 individual results for $\Omega(10)$ (small gray circles) obtained from individual hardware runs, a wider scattering can be observed when compared to the corresponding data for $\Omega(1 \mid 0)$ (left panel); the standard deviation (blue error bar) was found to be about three times larger, which is expected considering the significant increase in CX-gate applications. Nevertheless, the overall estimator for $\Omega(10)$ (blue circles) is in good agreement with the analytical and simulated values and is clearly distinguishable from the data for $\Omega(1 \mid 0)$ obtained from *IBMQ-27 Ehningen*.

TABLE I. Hardware requirements for the different variants of the Δ^λ subcircuits shown in Fig. 9.

Δ^λ-subcircuit		CX-gate applications		
variant	Qubits	All-to-all	*IBMQ-27*	Overhead
(a)	8	25	44	19
(b)	7	19	29	10
(c)	7	16	25	9

V. CONCLUSION

In this article, we introduced an amplitude-based encoding for an approximation of the USF. This involved a quantum circuit that, based on a given input, prepared a quantum register to reflect the corresponding USF output. Two circuit variations were suggested receiving the input either directly from a classical computer or via a quantum state when incorporated into a larger circuit. For the first variation, we demonstrated different levels of approximation and furthermore showed small circuit extension allowing to approximate other nonlinear functions. For the second variation we likewise presented a circuit extension for quantum-state inputs in superposition, e.g., when computing the average output of the USF. Supported by analytical and simulated data, the performance for all quantum circuits was evaluated on the IBM Quantum device in

Ehningen, Germany. Reliable results were presented, obtained from quantum circuits that included up to eight qubits, and up to 25 CX-gate applications, clearly demonstrating the applicability of our approach on state-of-the-art Noisy Intermediate-Scale Quantum devices available to date. We emphasize that only very recently the activation of a quantum neuron was successfully implemented by incorporating the single-step gearbox based on the RUS protocols, [51] and we are convinced that the concepts presented herein will be of significant influence for the future design of quantum neural network architectures. Finally, the amplitude-based encoding of nonlinearity may likewise be useful in quantum algorithms applied in different research areas, e.g., mathematical finance [52].

ACKNOWLEDGMENTS

This work was supported by the project AnQuC-3 of the Competence Center Quantum Computing Rhineland-Palatinate (Germany). We are highly indebted for being granted access to the IBM Quantum device in Ehningen, Germany. J.K. thanks Paul Nation from IBM for the helpful discussions about hardware-optimization strategies.

APPENDIX A: SIMULATION

The PYTHON code for constructing, optimizing, and executing the quantum circuits discussed in this article was written using the QISKIT framework [53]. All simulations were performed using the AERSIMULATOR included in QISKIT. Throughout, an ideal, noise-free quantum computer with all-to-all connectivity was assumed. We would like to emphasize that the number of circuit executions for each input angle or state has been set to 10^5 for maintaining comparability to the data obtained from *IBMQ-27 Ehningen*. However, sufficiently small confidence intervals were already observed for a significantly lower amount of repetitions, ranging between 2^{10} to 2^{14}.

APPENDIX B: HARDWARE-OPTIMIZATION PROCEDURE

Optimizing the quantum circuits demonstrated in this article on *IBMQ-27 Ehningen* requires transpilation prior to execution. Within QISKIT terminology, transpilation can be understood as a pipeline consisting of four steps: (1) rewriting the circuit in terms of the basis gate library of the backend; (2) mapping the virtual qubits to the physical qubits of the device (initial layout); (3) incorporating swap operations necessary due to the restricted connectivity of the involved physical qubits; and (4) optimizing the employed gates. Throughout we chose transpilation using the highest diligence (optimization level 3) that relies on the SWAP-based bidirectional heuristic search algorithm (SABRE) for finding the optimum initial layout and swapping strategy [54] and additionally perform full single- and double-qubit-gate optimizations. Due to its stochastic nature, we repeated transpilation 50 times, and saved the resulting transpiled circuit with the lowest number of CX gates. Note that the corresponding virtual-to-physical qubit mapping is optimum in terms of CX-gate applications, but does not consider the quality of the involved qubits. Since we only use 8 of the 27 qubits available on *IBMQ-27 Ehningen* at most, typically this transpiled circuit can be reconstructed using different qubit subsets. The MAPOMATIC package [55] allows to evaluate all of these different subsets regarding their individual error rates. In such a way we eventually identified the optimum hardware-realization of the circuit with respect to the CX-gate applications and the quality of the involved qubits. It is emphasized that the full procedure is only required once for a distinct circuit and does not need to be repeated when changing an input angle or state. Postmeasurement, error mitigation using the matrix-free measurement mitigation (M3) package was applied throughout [56]. Every circuit was executed 10^5 times, corresponding to the current maximum number of repetitions on the *IBMQ-27 Ehningen*. It should be noted that all hardware runs were conducted shortly after (typically less than 30 minutes) calibration of the device.

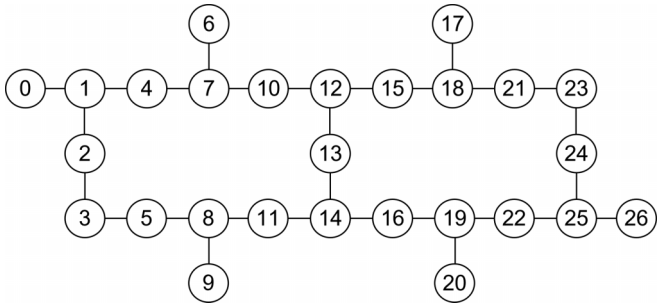

FIG. 10. Heavy-hexagon architecture of the *IBMQ-27 Ehningen* device.

APPENDIX C: HARDWARE CHARACTERISTICS

At the time of circuit execution, the *IBMQ-27 Ehningen* device was equipped with a Falcon r5.11 processor type. The average single gate error was about 2.9×10^{-4}, the average CX-gate error was determined to be about 10^{-2}. Average relaxation times were found at about 140 µs ($T_1 \sim 160$ µs and $T_2 \sim 110$ µs). Readout assignment errors had an average value of about 10^{-2}. Average CX-gate times were 300 ns. The exact characteristics of all employed qubits are detailed in the Supplemental Material in Table TS1. In Fig. 10, the typical heavy-hexagon architecture of the 27 qubits for *IBMQ-27 Ehningen* is demonstrated.

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

[2] S. C. Kak, Quantum neural computing, in *Advances in Imaging and Electron Physics*, edited by P. W. Hawkes (Elsevier, Amsterdam, 1995), Vol. 94, pp. 259–313.

[3] A. A. Ezhov and D. Ventura, Quantum neural networks, in *Future Directions for Intelligent Systems and Information Sciences: The Future of Speech and Image Technologies, Brain Computers, WWW, and Bioinformatics*, Studies in Fuzziness and Soft Computing, edited by N. Kasabov (Physica-Verlag HD, Berlin, 2000), pp. 213–235.

[4] M. Schuld, I. Sinayskiy, and F. Petruccione, The quest for a Quantum Neural Network, Quantum Inf. Process. **13**, 2567 (2014).

[5] K. H. Wan, O. Dahlsten, H. Kristjánsson, R. Gardner, and M. S. Kim, Quantum generalisation of feedforward neural networks, Npj Quantum Inf. **3**, 36 (2017).

[6] 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).

[7] R. Zhao and S. Wang, A review of quantum neural networks: Methods, models, dilemma, arXiv:2109.01840 (2021).

[8] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, The theory of variational hybrid quantum-classical algorithms, New J. Phys. **18**, 023023 (2016).

[9] G. R. Steinbrecher, J. P. Olson, D. Englund, and J. Carolan, Quantum optical neural networks, Npj Quantum Inf. **5**, 60 (2019).

[10] A. Skolik, J. R. McClean, M. Mohseni, P. van der Smagt, and M. Leib, Layerwise learning for quantum neural networks, Quantum Mach. Intell. **3**, 5 (2021).

[11] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, Ko. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, Variational quantum algorithms, Nat. Rev. Phys. **3**, 625 (2021).

[12] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, The power of quantum neural networks, Nat. Comput. Sci. **1**, 403 (2021).

[13] F. Rosenblatt, The perceptron: A perceiving and recognizing automaton, Technical Report No. 85-460-1 (Cornell Aeronautical Laboratory, Ithaca, New York, January 1957).

[14] S. Aaronson, Read the fine print, Nat. Phys. **11**, 291 (2015).

[15] A. Daskin, A simple quantum neural net with a periodic activation function, in *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan* (IEEE, New York, 2018), pp. 2887–2891.

[16] F. Tacchino, C. Macchiavello, D. Gerace, and D. Bajoni, An artificial neuron implemented on an actual quantum processor, Npj Quantum Inf. **5**, 26 (2019).

[17] Y. Cao, G. G. Guerreschi, and A. Aspuru-Guzik, Quantum neuron: An elementary building block for machine learning on quantum computers, arXiv:1711.11240.

[18] W. Hu, Towards a real quantum neuron, Natural Science **10**, 99 (2018).

[19] E. Torrontegui and J. J. García-Ripoll, Unitary quantum perceptron as efficient universal approximator, Europhys. Lett. **125**, 30004 (2019).

[20] S. Yan, H. Qi, and W. Cui, Nonlinear quantum neuron: A fundamental building block for quantum neural networks, Phys. Rev. A **102**, 052421 (2020).

[21] S. Chakraborty, T. Das, S. Sutradhar, M. Das, and S. Deb, An analytical review of quantum neural network models and relevant research, in *2020 5th International Conference on Communication and Electronics Systems (ICCES)* (IEEE, New York, 2020), pp. 1395–1400.

[22] J. Allcock, K. Hsieh, I. Kerenidis, and S. Zhang, Quantum Algorithms for Feedforward Neural Networks, ACM Trans. Quantum Comput. **1**, 1 (2020).

[23] F. Tacchino, P. Barkoutsos, C. Macchiavello, I. Tavernelli, D. Gerace, and D. Bajoni, Quantum implementation of an artificial feed-forward neural network, Quantum Sci. Technol. **5**, 044010 (2020).

[24] N. Miller and S. Mukhopadhyay, A quantum Hopfield associative memory implemented on an actual quantum processor, Sci. Rep. **11**, 23391 (2021).

[25] M. Maronese, C. Destri, and E. Prati, Quantum activation functions for quantum neural networks, Quantum Inf. Process **21**, 128 (2022).

[26] R. N. Bracewell, Heaviside's unit step function, h(x), in *The Fourier Transform and Its Applications, 3rd ed.*, (Cambridge University Press, Cambridge, England, 2000), pp. 61–65.

[27] G. Savaré, On the regularity of the positive part of functions, Nonlinear Anal. Theory Methods Appl. **27**, 1055 (1996).

[28] R. Korn, E. Korn, and G. Kroisandt, *Monte Carlo Methods and Models in Finance and Insurance*, Chapman and Hall/CRC Financial Mathematics Series (Taylor & Francis, London, 2010).

[29] P. Rebentrost, B. Gupt, and T. R. Bromley, Quantum computational finance: Monte Carlo pricing of financial derivatives, Phys. Rev. A **98**, 022321 (2018).

[30] S. Woerner and D. J. Egger, Quantum risk analysis, Npj Quantum Inf. **5**, 15 (2019).

[31] A. C. Vazquez and S. Woerner, Efficient State Preparation for Quantum Amplitude Estimation, Phys. Rev. Appl. **15**, 034027 (2021).

[32] N. Stamatopoulos, D. J. Egger, Y. Sun, C. Zoufal, R. Iten, N. Shen, and S. Woerner, Option Pricing using Quantum Computers, Quantum **4**, 291 (2020).

[33] S. Chakrabarti, R. Krishnakumar, G. Mazzola, N. Stamatopoulos, S. Woerner, and W. J. Zeng, A threshold for quantum advantage in derivative pricing, Quantum **5**, 463 (2021)

[34] N. Wiebe and V. Kliuchnikov, Floating point representations in quantum circuit synthesis, New J. Phys. **15**, 093041 (2013).

[35] A. Paetznick and K. M. Svore, Repeat-until-success: Non-deterministic decomposition of single-qubit unitaries, Quantum Inf. Comput. **14**, 1277 (2014).

[36] A. Bocharov, M. Roetteler, and K. M. Svore, Efficient Synthesis of Universal Repeat-Until-Success Quantum Circuits, Phys. Rev. Lett. **114**, 080502 (2015).

[37] N. Wiebe and M. Rötteler, Quantum arithmetic and numerical analysis using repeat-until-success circuits, Quantum Inf. Comput. **16**, 134 (2016).

[38] G. G. Guerreschi, Repeat-until-success circuits with fixed-point oblivious amplitude amplification, Phys. Rev. A **99**, 022306 (2019).

[39] V. Nair and G. E. Hinton, Rectified linear units improve restricted boltzmann machines, in *ICML* (Omnipress, Madison, WI, 2010), pp. 807–814.

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

[41] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevA.107.022606 for a more technical analysis, transpiled quantum circuits as well as the hardware characteristics of the used quantum computer.

[42] J. Han and C. Moraga, The influence of the sigmoid function parameters on the speed of backpropagation learning, in *From Natural to Artificial Neural Computation*, edited by J. Mira and F. Sandoval (Springer, New York, 1995), pp. 195–201.

[43] M. Möttönen, J. J. Vartiainen, V. Bergholm, and M. M. Salomaa, Quantum Circuits for General Multiqubit Gates, Phys. Rev. Lett. **93**, 130502 (2004).

[44] V. V. Shende, S. S. Bullock, and I. L. Markov, Synthesis of quantum-logic circuits, IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. **25**, 1000 (2006).

[45] J. Li and S. Kais, A universal quantum circuit design for periodical functions, New J. Phys. **23**, 103022 (2021).

[46] T. Toffoli, Reversible computing, in *Automata, Languages and Programming*, edited by Jaco de Bakker and Jan van Leeuwen (Springer, Berlin, 1980), pp. 632–644.

[47] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition* (Cambridge University Press, Cambridge, England, 2010).

[48] V. V. Shende and I. L. Markov, On the cnot-cost of toffoli gates, Quantum Inf. Comput. **9**, 461 (2009).

[49] D. P. DiVincenzo and J. Smolin, Results on two-bit gate design for quantum computers, in *Proceedings Workshop on Physics and Computation. PhysComp '94* (IEEE, New York, 1994), pp. 14–23.

[50] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, Elementary gates for quantum computation, Phys. Rev. A **52**, 3457 (1995).

[51] M. S. Moreira, G. G. Guerreschi, W. Vlothuizen, J. F. Marques, J. van Straten, S. P. Premaratne, X. Zou, H. Ali, N. Muthusubramanian, C. Zachariadis, J. van Someren, M. Beekman, N. Haider, A. Bruno, C. G. Almudever, A. Y. Matsuura, and L. DiCarlo, Realization of a quantum neural network using repeat-until-success circuits in a superconducting quantum processor, arXiv:2212.10742.

[52] M.-O. Wolf, R. Horsky, and J. Koppe, A quantum algorithm for pricing asian options on valuation trees, Risks **10**, 221 (2022).

[53] M. D. Sajid Anis, H. Abraham, A. Offei, R. Agarwal, G. Agliardi, M. Aharoni, I. Y. Akhalwaya, G. Aleksandrowicz, T. Alexander, M. Amy, S. Anagolum, E. Arbel, A. Asfaw, A. Athalye, A. Avkhadiev *et al.*, QISKIT: An open-source framework for quantum computing (2021).

[54] G. Li, Y. Ding, and Y. Xie, Tackling the qubit mapping problem for nisq-era quantum devices, in *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems, ASPLOS '19* (ACM, New York, 2019), pp. 1001–1014.

[55] P. Nation and M. Treinish, MAPOMATIC: QISKIT partners, 2022, https://github.com/Qiskit-Partners/mapomatic.

[56] P. Nation, H. Kang, N. Sundaresan, and J. M. Gambetta, Scalable mitigation of measurement errors on quantum computers, PRX Quantum **2**, 040326 (2021).