Supplementary Information to: An Artificial Neuron Implemented on an Actual Quantum Processor

Francesco Tacchino,[1,*] Chiara Macchiavello,[1,†] Dario Gerace,[1,‡] and Daniele Bajoni[2,§]

[1] *Dipartimento di Fisica, Università di Pavia, via Bassi 6, I-27100, Pavia, Italy*
[2] *Dipartimento di Ingegneria Industriale e dell'Informazione, Università di Pavia, via Ferrata 1, I-27100, Pavia, Italy*

I. ACTION OF THE U_w UNITARY OPERATOR

Here we give an intuitive geometrical representation of the action of the unitary operation U_w on the state vectors. In Fig. S1 we schematically represent $|\psi_i\rangle$ and $|\psi_w\rangle$ as vectors in the m-dimensional Hilbert space of the computational basis to help visualizing the effect of applying U_w on $|\psi_i\rangle$. In this geometrical space, the inner product $\vec{i}\cdot\vec{w}$, i.e. the perceptron's operation, consists in the projection of vector $|\psi_i\rangle$ over the vector $|\psi_w\rangle$ (shown in green), up to a normalization factor. In particular, U_w rotates $|\psi_w\rangle$ such that it aligns with the axis given by $|111\ldots11\rangle$, and rotates $|\psi_i\rangle$ by the same angles. This means that after applying U_w to $|\psi_i\rangle$ the inner product $\vec{i}\cdot\vec{w}$ corresponds to the projection of the vector $U_w|\psi_i\rangle$ over the axis $|111\ldots11\rangle$.

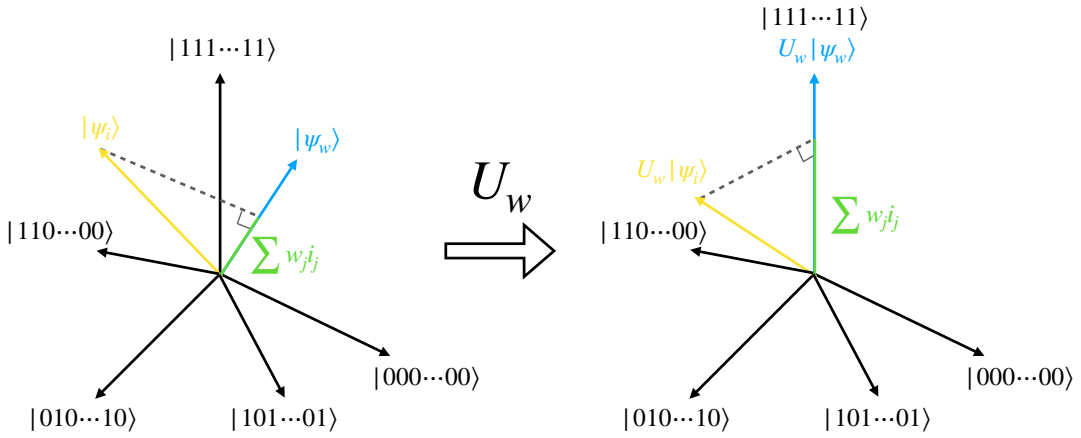

FIG. S1: Geometrical representation of the action of U_w on $|\psi_i\rangle$ and $|\psi_w\rangle$ in the m-dimensional Hilbert space of the N qubits.

II. HYPERGRAPH STATES

In this section, we provide a brief introduction to the class of hypergraph states, which we have employed in the main text as a key ingredient of the quantum algorithm for the quantum perceptron modeling of an artificial neuron. A more complete (and technical) discussion can be found in Refs. 1, 2, where alternative definitions using the stabilizer formalism are also given.

*Electronic address: francesco.tacchino01@ateneopv.it
†Electronic address: chiara.macchiavello@unipv.it
‡Electronic address: dario.gerace@unipv.it
§Electronic address: daniele.bajoni@unipv.it

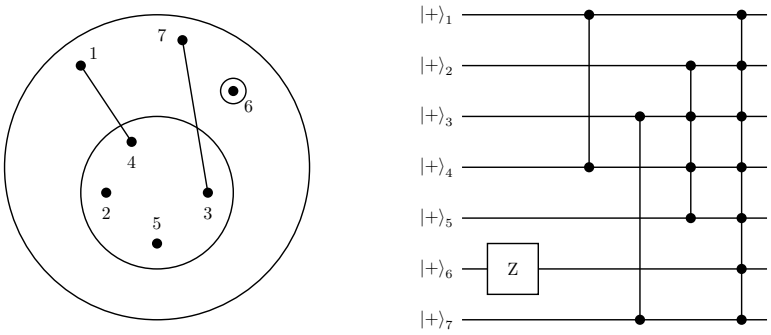

FIG. S2: A mathematical hypergraph, with hyper-edges represented as circles around subsets of vertices; on the right, the quantum circuit generating the corresponding hypergraph state is explicitly shown. Notice that, since all the operations commute with each other, the sequence of gates can be performed in any order.

We define a k-uniform hypergraph $g_k = \{V, E\}$ as a collection of n vertices V with a set of k-hyper-edges E, where each k-hyper-edge connects exactly k vertices, and the usual notion of a connected graph is obtained for $k = 2$. The k-uniform hypergraph state associated to g_k is obtained in the following way: after assigning a qubit in the $\sqrt{2}|+\rangle = |0\rangle + |1\rangle$ state to each vertex, for any k-hyper-edge a (multi)-controlled Z operation is performed between all the qubits $\{i_1, \ldots, i_k\}$ connected by that hyper-edge. We thus obtain

$$|g_k\rangle = \prod_{\{i_1,\ldots,i_k\}\in E} \mathrm{C}^k\mathrm{Z}_{i_1,\ldots,i_k}|+\rangle^{\otimes n}\,. \tag{1}$$

A hypergraph $g_{\leq n} = \{V, E\}$ is a set of n vertices V with a set E of hyper-edges of any order k, not necessarily uniform. A quantum state of n qubits can be associated to any hypergraph $g_{\leq n}$ by generalizing the same procedure given above for the k-uniform case, namely by performing a $\mathrm{C}^k\mathrm{Z}$ gate for each hyper-edge in $g_{\leq n}$

$$|g_{\leq n}\rangle = \prod_{k=1}^{n} \prod_{\{i_1,\ldots,i_k\}\in E} \mathrm{C}^k\mathrm{Z}_{i_1,\ldots,i_k}|+\rangle^{\otimes n}\,. \tag{2}$$

It is immediately clear that k-uniform hypergraph states form a subset of the class of hypergraph states. Moreover, for n qubits there are exactly $\mathcal{N} = 2^{2^n-1}$ different hypergraph states. In Fig. S2 we show an example of a hypergraph, with the quantum circuit generating its corresponding state.

From a formal point of view, the encoding scheme proposed in the main text assigns a boolean value representing a single pixel to each element of an n-qubit computational basis. This value is converted into a $\pm$ sign, i.e. a 0 or π phase, in front of each component in a balanced quantum superposition of all the computational basis elements. The quantum states resulting from this scheme are known as Real Equally Weighted (REW). Given a boolean function $f : \{0,1\}^n \to \{0,1\}$, the corresponding REW state is

$$|f\rangle = \frac{1}{2^{n/2}} \sum_{x=0}^{2^n-1} (-1)^{f(x)}|x\rangle\,. \tag{3}$$

The connection between the REW states, naturally appearing in our encoding based on hypergraph states, is provided by the following general result [1]: *the set $G_\pm$ of REW states and the set $G_{\leq n}$ of hypergraph states coincide.* Indeed, $G_{\leq n} \subseteq G_\pm$ since any $|g_{\leq n}\rangle$ is obtained from the REW $|+\rangle^{\otimes n}$ by applying (multi)-controlled-Z (i.e. π phases) operations. The opposite inclusion can be proven constructively by applying the inverse of the algorithm considered in the main text for the generation of hypergraph states (see again Ref. 1 for details).

As a consequence of the results above, there are exactly $\mathcal{N}$ different REW states, which can be mapped into $\mathcal{N}$ hypergraphs corresponding to a precise generation algorithm involving (multi)-controlled-Z gates. Given that a pattern and its negative are mapped into the same quantum state as a consequence of global phase symmetry, under the encoding scheme proposed in the main text $2\mathcal{N}$ is the number of distinct inputs or weights that can be analyzed by the quantum artificial neuron algorithm running on n qubits.

III. GATE IDENTITIES

Here we show some standard textbook [3] identities that were used to decompose the algorithms presented in the main text, for completeness. In Fig. S3, special attention is given to the use of single- and two-qubit gates natively available on the IBM-Q hardware (single qubit rotations and CNOT). On the real processors, a further layer of complexity is introduced by limitations in the connectivity between the different qubits, the CNOT being allowed only between specific pairs of qubits and with given control-target relations.

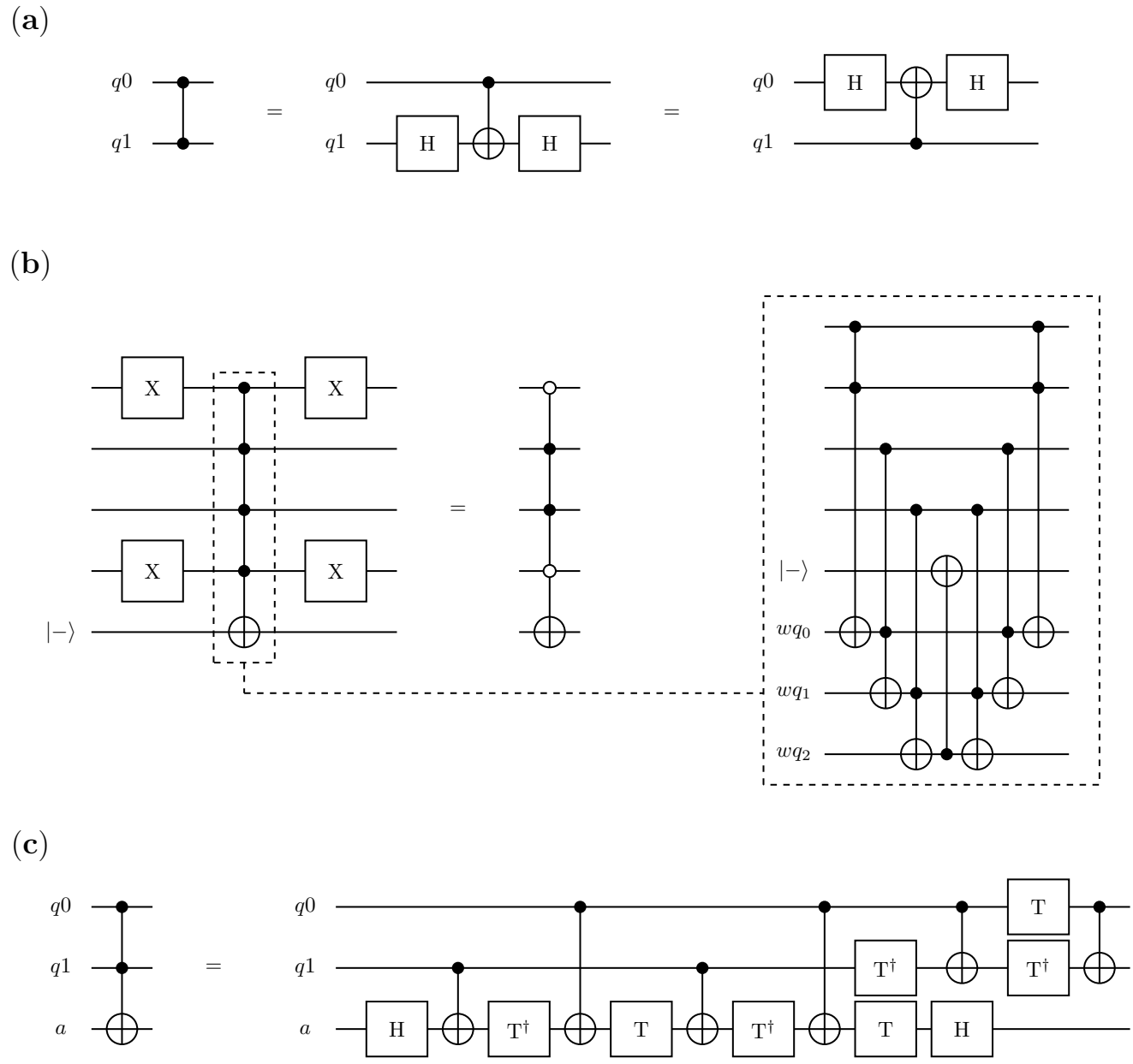

FIG. S3: Elementary gate decompositions. (a) CZ from CNOT and Hadamard. (b) An example of a multi-controlled sign flip block $\mathrm{SF}_{4,6}$ adding a -1 factor in front of the $|0110\rangle$ component of a 4-qubit computational basis, making use of an ancilla in the $\sqrt{2}|-\rangle = |0\rangle - |1\rangle$ state. In each block, the NOT gates should be operated on those qubits for which the state is $|0\rangle$ in the target component. A standard construction to implement the multi-controlled $\mathrm{C}^N\mathrm{NOT}$ gate using $\mathrm{N}-1$ additional working qubits is also shown. (c) Textbook decomposition of a Toffoli gate using CNOT and single qubit gates.

IV. ADDITIONAL DATA

In this section, we present some additional results from the simulation of selected input-weight cases for $N = 4$ qubits, performed on the IBM Qiskit simulator. As reported in the main text, the $\vec{i}$ and $\vec{w}$ vectors can be interpreted as 4×4 patterns. After setting a threshold at $O(i, w) = |\sum_j i_j w_j|^2 > 0.5$ and choosing a cross-shaped $\vec{w}_t$ vector, in Fig. S4 we show some possible positive matches differing from the weight pattern (or from its negative) by at most two pixels. In Fig. S5 we report a selection of inputs with a negative output, i.e. not triggering the activation of the artificial neuron threshold function. For each pattern, the algorithm and the final readout of the output qubit were repeated $n = 8192$ times to estimate the probabilities of the $|0\rangle$ and $|1\rangle$ components: the discrepancies between the computed (i.e., ideally predicted with standard linear algebra calculations) and the quantum simulated values for each output $O(i, w) = |c_{m-1}|^2$ are solely due to the finite statistics of this final measure.

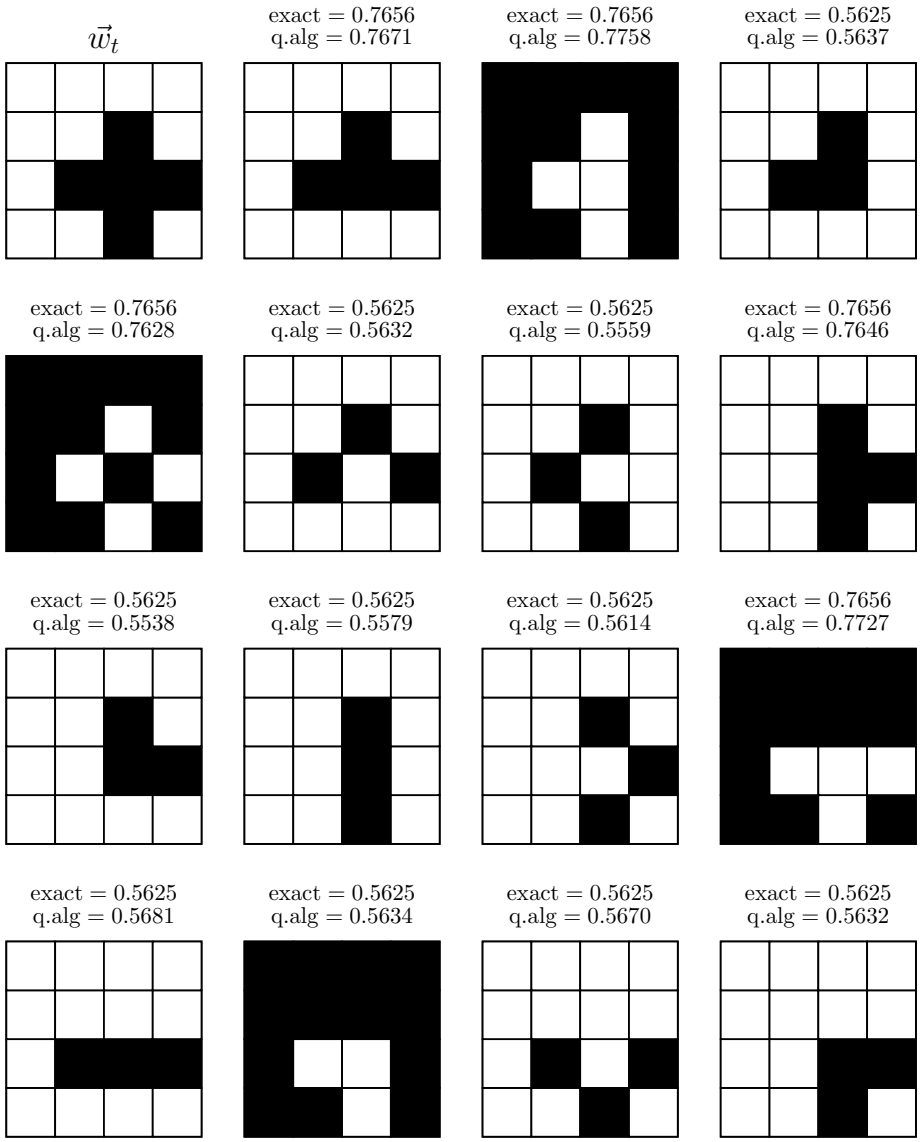

FIG. S4: Weight and positive input patterns for the $N = 4$ case. The negative of each pattern would also give the same result as the one reported here. Above each input pattern, the quantitative answers of the artificial neuron, namely the values of $|c_{m-1}|^2$, are reported as obtained either through standard linear algebra (ideal 'exact' results) or resulting from the simulation of the quantum algorithm ('q. alg', run on a classical computer, with $n_{shots} = 8192$ repetitions). The two versions agree within statistical error.

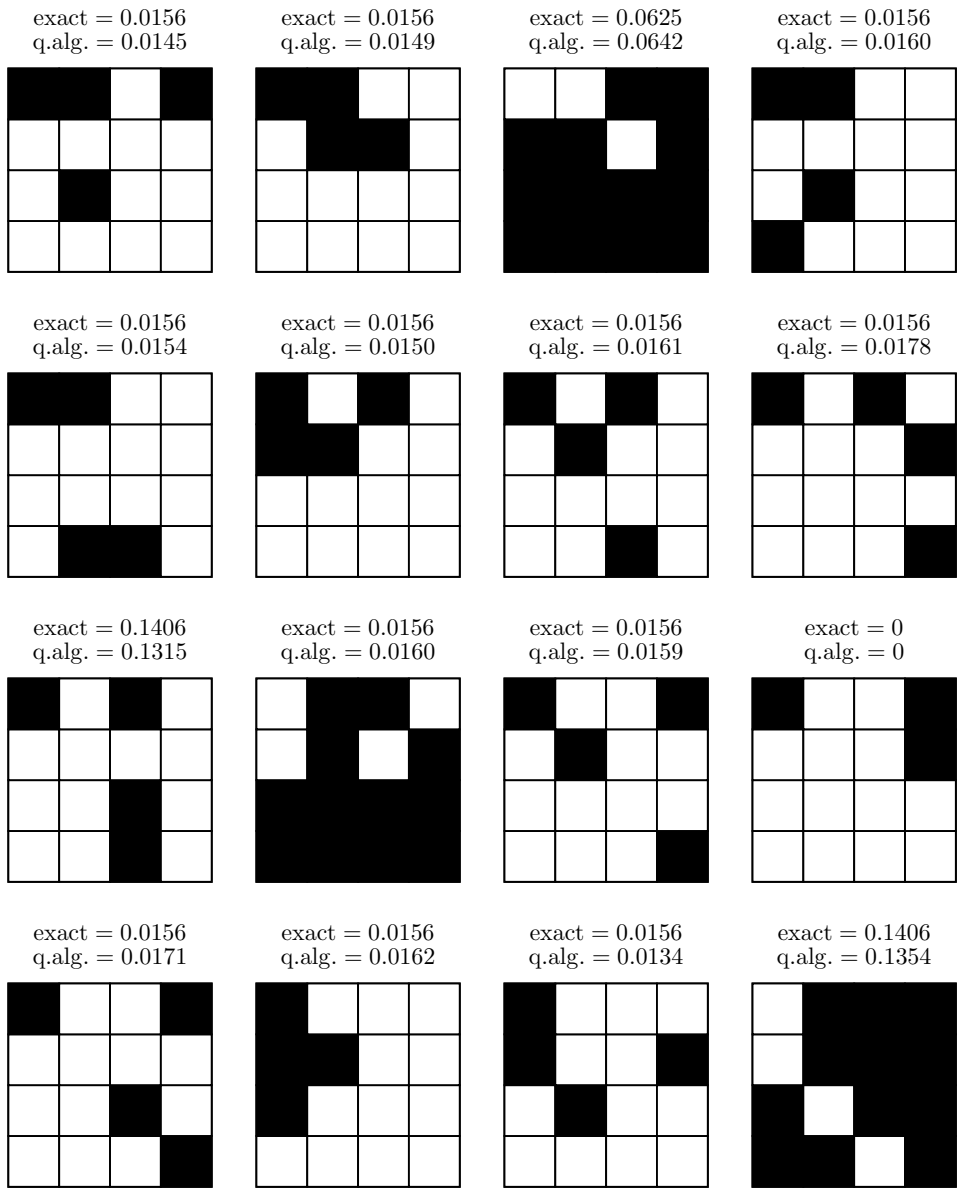

FIG. S5: Selection of negative input patterns for the $N = 4$ result and for the same weight vector as in Fig. S4. Also in this case, both a given pattern and its negative would yield the same result and the 'exact' and 'q. alg' labels indicate the values of $|c_{m-1}|^2$ as obtained either through standard linear algebra or from the simulation of the quantum algorithm.

[1] M. Rossi, M. Huber, D. Bruß and C. Macchiavello, *Quantum hypergraph states*, New Journal of Physics **15**, 113022 (2013).
[2] M. Ghio, D. Malpetti, M. Rossi, D. Bruß and C. Macchiavello, *Multipartite entanglement detection for hypergraph states*, J. Phys. A: Math. Theor. **51**, 045302 (2018).
[3] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, UK, 2000).