ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

Supervised quantum gate "teaching" for quantum hardware design

Leonardo Banchi[1], Nicola Pancotti[2] and Sougato Bose[1] *

1- Department of Physics and Astronomy, University College London,
Gower Street, London WC1E 6BT, United Kingdom

2- Max-Planck-Institut für Quantenoptik,
Hans-Kopfermann-Straße 1, 85748 Garching, Germany

Abstract. We show how to train a quantum network of pairwise interacting qubits such that its evolution implements a target quantum algorithm into a given network subset. Our strategy is inspired by supervised learning and is designed to help the physical construction of a quantum computer which operates with minimal external classical control.

1 Quantum networks for computation

A quantum computer is a device which uses peculiar quantum effects, such as superposition and entanglement, to process information [1]. Although current quantum computers operate on few quantum bits (*qubits*) [2, 3], it is known that large-scale quantum devices can run certain algorithms exponentially faster than the classical or probabilistic counterpart [1].

Recently there has been many proposals to speed-up machine learning strategies using quantum devices [4]. Most of these proposals either use quantum algorithms to achieve faster learning [5] or exploit quantum fluctuations to escape from local minima in training the Boltzmann machine [6]. In this paper we consider a different perspective. Since the actual development of a general quantum computer is still in its infancy, rather than focusing on the advantages of quantum devices for machine learning we show how machine learning can help the construction of a quantum computer.

To keep the discussion realistic, we focus on a superconducting quantum computing architecture [7, 8] where each qubit is realized with a superconducting circuit cooled at low temperature. The pairwise coupling between two qubits is introduced by connecting them via a capacitor (or an inductor) whose strength can be tuned by design. Given the flexibility in wiring the different qubits, it is then possible to build a quantum network with tunable couplings.

In the next sections we introduce basic aspects of the physical simulation of quantum operations to set up the formalism and then we propose our strategy which is inspired by supervised learning.

*L.B. and S.B. acknowledge the financial support by the ERC under Starting Grant 308253 PACOMANEDIA.

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

1.1 Physical implementation of quantum gates

From the mathematical point of view each qubit is described by a two dimensional Hilbert space $\mathbb{C}^2$, while the Hilbert space of N qubits is given the the tensor product $\mathbb{C}^2 \otimes \mathbb{C}^2 \cdots = \mathbb{C}^{2^N} \equiv \mathcal{H}_N$. The possible quantum states, like the state 00101 for a classical 5-bit register, correspond to vectors of unit norm in the Hilbert space. An arbitrary operation, namely a quantum *gate*, corresponds to a unitary matrix U acting on $\mathcal{H}_N$. In more physical terms, U is the solution of the Schrödinger equation $i\frac{\partial U}{\partial t} = HU$, where i is the imaginary unit, t represents time, and H is the Hamiltonian, a Hermitian $2^N \times 2^N$ matrix which describes the physical interactions between the qubits. If the qubits are unmodulated, namely there is no external time-dependent control so H is independent on t, then after a certain time t the operation on the qubits is given by $U = e^{-itH}$, being $e^{(\cdot)}$ the matrix exponential. In principle, for any given operation U there are some corresponding interactions modeled by a Hamiltonian H so that $U = e^{-itH}$. However, in physical implementations of quantum computers [2] the range of possible Hamiltonians is severely limited, thus drastically restricting the range of achievable quantum operations without external control. This problem is typically solved by switching on and off different interactions in time so the final operation is the product $U_1 U_2 \ldots$ where $U_n = e^{-it_n H_n}$, being H_n a sequence of interaction Hamiltonians, each one switched on for a time t_n. As in classical computation, there is a minimal set of gates $\{U_n\}$ which enable *universal* quantum computation simply by concatenating at different times gates from this set [9]. Indeed, most quantum algorithms are nothing but a known sequence of universal operations. The implementation of this sequence however requires an outstanding experimental ability to perfectly switch on and off different physical couplings at given times. Possible errors or imperfections in this sequential process accumulate in time and may affect the outcome, if not tacked with error correcting codes [1].

Given this experimental difficulty, it is worth asking whether the unitary U which results from a recurring sub-sequence of the algorithm can be implemented directly in *hardware*. Indeed in the next section we propose a different strategy which exploits auxiliary qubits to implement quantum operations with physical interactions and no external control. This strategy could potentially allow an experimentalist to create a quantum device which, by simply "waiting" for the natural dynamics of the network, is able to implement transformations like the Quantum Fourier Transform which are ubiquitous in quantum algorithms, and may even provide an alternative paradigm for general quantum computation. Our method shares similar goals with a recent proposal by Childs [10], but it is completely different because it uses weighted networks which allow us to significantly lower the number ancillary qubits required for the operation.

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

1.2 Engineered unmodulated networks for computation

Our aim is to implement a quantum operation (a unitary[1] matrix U) on a N qubit register exploiting the physical interactions available for that hardware architecture, and avoiding to use external control fields. Since a generic $2^N \times 2^N$ gate U is impossible to realize with N qubits and pairwise interactions only, as described before, we consider a larger network of $N' > N$ qubits and we engineer the strength of the pairwise interactions between them to implement the operation, when possible.

This problem shares some similarities with supervised learning in artificial neural networks although, as we clarify in the following, is also very different in some aspects. In supervised learning, given a training set $\{I_k, O_k\}_{k=1,\dots,M}$ the goal is the find a functional approximation $O_k = f(I_k)$ which is also able to predict the output corresponding to unknown inputs missing from the training set. Even when there is no prior knowledge of f, it is possible to approximate the input/output relations with a neural network composed by input, output and hidden layers [11]. The learning procedure then consists in finding the optimal weights between nodes of different layers such that the desired input/output relation is reconstructed.

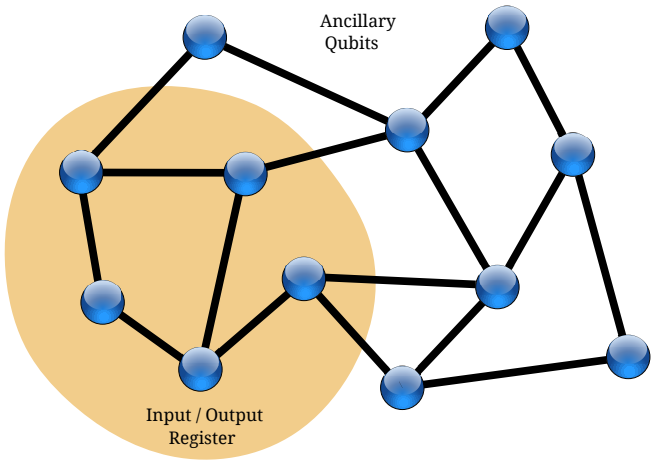

Fig. 1: Example of a quantum network composed of register and ancillary qubits. Each vertex is a qubit and each edge represents pairwise interactions. The target operation U is implemented only on the register qubits.

On the other hand, in our problem the functional relation between inputs and outputs, namely the gate U is already known in advance. However, the corresponding Hamiltonian may contain simultaneous interactions between 3 or more qubits which unlikely appear in physical implementations. To simulate U

[1] We consider a quantum gate, but our formalism can be easily extended to more general quantum channels [1].

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8.
Available from http://www.i6doc.com/en/.

using pairwise interactions only, we consider then a larger network as in Fig. 1 with ancillary qubits playing the role of the hidden layers, and we train the weights, namely the physical couplings, such that the dynamics of the network reproduces U in a given subset of qubits. To simplify the implementation we assume that the input and output registers are made of the same physical qubits, although this assumption may be removed. In the next section we show how to formally model the training procedure.

2 Supervised gate "teaching"

In supervised learning the training set is composed of data, whose input to output map f is not known. On the other hand, we know the gate U and we want find find the physical couplings, i.e. the weights of the network in Fig. 1, such that the quantum network evolution implements U in the register. Because of this difference we named our strategy "teaching" rather than learning. In our case we can build an arbitrary large training set by choosing random input states[2] $|\psi_j\rangle \in \mathcal{H}_N$ and finding the corresponding output states $\mathcal{H}_N \ni |\psi'_j\rangle = U|\psi_j\rangle$, so the generated M-dimensional (M being variable) training set is

$$\mathcal{T} = \{(|\psi_j\rangle, U|\psi_j\rangle) \quad : \quad j = 1, \ldots, M\} \ . \tag{1}$$

In principle the optimal inputs may depend on the target gate U. However, for simplicity and generality, in (1) each $|\psi_i\rangle$ is sampled from the Haar measure [12].

The quality of the implementation of the target gate U into the dynamics of the quantum network can be measured by defining a *cost* function which, for any input state $|\psi_j\rangle \in \mathcal{T}$, measures the distance between the output state of the evolution and the expected output $U|\psi_j\rangle$. The similarity between two quantum states $|\psi\rangle$ and $|\phi\rangle$ is measured by the fidelity[3] $|\langle\psi|\phi\rangle|^2$. Because of the Cauchy-Schwarz inequality, it is $0 \leq |\langle\psi|\phi\rangle|^2 \leq 1$ and the upper value $|\langle\psi|\phi\rangle| = 1$ is obtained only when $|\psi\rangle = |\phi\rangle$. However, because of entanglement between the register and the ancillary qubits the state of the register is not exactly known, but it can be in different states $|\phi_j\rangle$ with probability p_j. Such a *mixed* state is mathematically described by the matrix [1] $\rho = \sum_j p_j |\phi_j\rangle\langle\phi_j|$ and the fidelity between ρ and ψ becomes $\langle\psi|\rho|\psi\rangle \equiv \sum_j p_j |\langle\psi|\phi_j\rangle|^2$.

Given an initial state $|\psi\rangle$ let us call $\mathcal{E}_w[\psi]$ the state of the register after the evolution generated by the interaction Hamiltonian $H(w)$ which depends on the weights w. This state can be constructed by (i) initializing the N register qubits in the state $|\psi\rangle$; (ii) setting the remaining $N' - N$ ancillary qubits in a (fixed) state $|\alpha\rangle$; (iii) switching on the evolution described by $H(w)$ for a certain time t – without loss of generality we set $t = 1$, since $e^{-itH(w)} = e^{-iH(tw)}$; (iv) observing the state of the register[4]. The goal is then to find the optimal weights w (if

[2] In the "bra and ket" notation [1] $|\psi\rangle$ refers to a normalized vector.

[3] The "bra" $\langle\psi|$ corresponds to the Hermitian conjugate $|\psi\rangle^\dagger$ so $\langle\psi|\phi\rangle$ is the inner product between the vectors $|\psi\rangle$ and $|\phi\rangle$.

[4] Formally, steps (i) and (ii) correspond to the preparation of the state $|\eta_0\rangle = |\psi\rangle \otimes |\alpha\rangle$. Step (iii) gives the state $|\eta_t\rangle = e^{-itH(w)}|\eta_0\rangle$. While step (iv) produces a mixed state since the dynamics of a reduced system is obtained with the *partial trace* [1], so $\mathcal{E}_w[\psi] = \mathrm{Tr}_{\mathrm{ancilla}} |\eta_t\rangle\langle\eta_t|$.

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

they exist) such that the state $\mathcal{E}_w[\psi_j]$ is equal to the target state $U|\psi_j\rangle$ for each $|\psi_j\rangle \in \mathcal{T}$. Mathematically this corresponds to the maximization of the average fidelity

$$F(w) = \frac{1}{M} \sum_{|\psi_j\rangle \in \mathcal{T}} \langle \psi_j | U^\dagger \mathcal{E}_w[\psi_j] U |\psi_j\rangle \ . \tag{2}$$

The function $F(w)$ measures on average how the dynamics of the network reproduces in the register the target quantum gate U. It is non-convex in general and may have many local maxima with $F < 1$. However, an optimal configuration $\tilde{w}$ is interesting for practical purposes only if the error $\epsilon = 1 - F(\tilde{w})$ is smaller than the desired threshold [13] (say 10^{-3} – 10^{-4}), so in the following we say that a solution exists if an optimal $\tilde{w}$ is found with $\epsilon < 10^{-3}$. There are no known theoretical tools to establish in advance whether this high-fidelity solution $\tilde{w}$ can exist, so one has to rely on numerical methods. In the next section we discuss a simple algorithm for finding $\tilde{w}$.

2.1 A simple algorithm for network optimization

The explicit form of the fidelity function ($1 - F$ is a cost function) as a sum over the training set allows us to use the stochastic gradient descent training algorithm, which is widely used for in training artificial neural networks within the backpropagation algorithm. However, since our training set is variable and can be sampled from the Haar distribution of pure states we propose an adapted version of the stochastic gradient descent with online training:

1: Choose the initial weights w (e.g. at random);
2: choose an initial learning rate κ;
3: **repeat**
4: generate a random $|\psi\rangle$ from the Haar measure;
5: **for** $j = 1, \ldots, L$ **do**
6: update the weights as

$$w \to w + \kappa \nabla_w \langle \psi | U^\dagger \mathcal{E}_w [\psi] U |\psi\rangle \, ; \tag{3}$$

7: **end for**
8: decrease κ (see below);
9: **until** convergence (or maximum number of operations).

In the above algorithm, the weights are updated L times before changing the state. The parameter L defines the amount of deterministic steps in the learning procedure and it can be set to the minimum value 1, so that after each iteration the state is changed, or to higher values. On the other hand, the learning rate κ has to decrease [14] in an optimal way to assure convergence, a common choice being $\kappa \propto s^{-1/2}$ where s is the step counter. On physical grounds, as we discussed extensively in Ref. [12], the stochastic fluctuations given by choosing random quantum states at different steps enable the training procedure to escape from local maxima when the weights are far from the optimal point $\tilde{w}$.

Using the above algorithm, in Ref.[12] we considered pairwise interactions described by physically reasonable Hamiltonians and we found different quantum

ESANN 2016 proceedings, European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning. Bruges (Belgium), 27-29 April 2016, i6doc.com publ., ISBN 978-287587027-8. Available from http://www.i6doc.com/en/.

network configurations which implement different quantum operations, such as the quantum analogue of Toffoli and Fredkin gate.

3 Concluding remarks

We are proposing an alternative strategy to the physical implementation of quantum operations which avoids time modulation and sophisticated control pulses. This strategy consists in enlarging the number of qubits and engineering the unmodulated pairwise interactions between them so that the desired operation is implemented into the register subset (see Fig. 1) by the natural physical evolution. Inspired by the analogy with the training of artificial neural networks, we then propose a simple algorithm that is suitable for finding few-qubits networks which implement some important quantum gates. In the long term, by finding more efficient training algorithms suitable for larger spaces, our strategy could potentially provide an alternative paradigm for computation where some quantum algorithms and/or many-qubit gates are obtained by simply "waiting" for the natural dynamics of a suitably designed network.

References

[1] M. A. Nielsen and I. L. Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2000.

[2] T. D. Ladd et al. Quantum computers. *Nature*, 464(7285):45–53, 2010.

[3] V. C. Coffey. The incremental quest for quantum computing. *Photonics Spectra*, 48(6):36–41, 2014.

[4] P. Wittek. *Quantum machine learning: what quantum computing means to data mining*. Elsevier, Oxford, 2014.

[5] P. Rebentrost, M. Mohseni, and S. Lloyd. Quantum support vector machine for big data classification. *Phys. Rev. Lett.*, 113(13):130503, 2014.

[6] M. Denil and N. De Freitas. Toward the implementation of a quantum RBM. In *NIPS Deep Learning and Unsupervised Feature Learning Workshop*, 2011.

[7] M. Mariantoni et al. Implementing the quantum von Neumann architecture with superconducting circuits. *Science*, 334(6052):61–65, 2011.

[8] Y. Chen et al. Qubit architecture with high coherence and fast tunable coupling. *Phys. Rev. Lett.*, 113(22):220502, 2014.

[9] A. Barenco et al. Elementary gates for quantum computation. *Phys. Rev. A*, 52(5):3457, 1995.

[10] A. M. Childs, D. Gosset, and Z. Webb. Universal computation by multiparticle quantum walk. *Science*, 339(6121):791–794, 2013.

[11] C. M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[12] L. Banchi, N. Pancotti, and S. Bose. Quantum gate learning in engineered qubit networks: Toffoli gate with always-on interactions. *arXiv preprint arXiv:1509.04298*, 2015.

[13] J. M. Martinis. Qubit metrology for building a fault-tolerant quantum computer. *Npj Quantum Information*, 1:15005, 2015.

[14] J. C. Spall. *Introduction to stochastic search and optimization: estimation, simulation, and control*, volume 65. John Wiley & Sons, 2005.