Quantum Information and Computation, Vol. 13, No. 1&2 (2013) 0036–0053

MULTIQUBIT ENTANGLEMENT OF A GENERAL INPUT STATE

ELIZABETH C. BEHRMAN
Department of Mathematics and Physics, Wichita State University
Wichita, KS 67260-0033,USA

JAMES E. STECK
Department of Aerospace Engineering, Wichita State University
Wichita, KS 67260-0044, USA

Received August 13, 2011
Revised July 30, 2012

Measurement of entanglement remains an important problem for quantum information. We present the design and simulation of an experimental method for an entanglement indicator for a general multiqubit state. The system can be in a pure or a mixed state, and it need not be "close" to any particular state. The system contains information about its own entanglement; we use dynamic learning methods to map this information onto a single experimental measurement which is our entanglement indicator. Our method does not require prior state reconstruction or lengthy optimization. An entanglement witness emerges from the learning process, beginning with two-qubit systems, and extrapolating this to three, four, and five qubit systems where the entanglement is not well understood. Our independently learned measures for three-qubit systems compare favorably with known entanglement measures. As the size of the system grows the amount of additional training necessary diminishes, raising hopes for applicability to large computational systems.

Keywords: quantum algorithm, entanglement, dynamic learning

Communicated by: R Jozsa & B Terhal

1 Introduction

Because entanglement is crucial to most quantum information applications, convenient and accurate methods for its measurement and calculation are essential. Alas, though considerable work by a large number of very intelligent people has made considerable progress (see, e.g., [1-12]), we have as yet no general method, of universal applicability, much less one which is easy to implement experimentally.

For 2-qubit systems the problem is fairly well understood, from more than one point of view. For example, the "entanglement of formation"[1, 2] approach is based on the idea of construction: how much entanglement is necessary in order to reconstruct this particular state? This gives us a general measure as an analytic formula, so that, if one knows the density matrix, one can easily calculate the entanglement. However, generalizations to larger systems are difficult[3]. Various geometric methods[4, 5] use an extremely appealing idea of "closeness" to the subspace of separable states: one constructs some distance measure

in the state space, then, performs a minimization calculation to find the distance of closest approach. Product states, of course, have zero distance to that space, and have therefore zero entanglement. Many entanglement witnesses[6] are based on this same idea. The drawback is, of course, the optimization calculation, which rapidly becomes impractically difficult for large systems. In addition, these approaches assume that one knows the density matrix. One might not. And reconstruction of the density matrix, like optimization, is not easy to generalize to large systems. Most witnesses also require that the state of the system be "close", in some sense, to a given, known state[7].

Clearly it would be of great interest to have a general, experimental method for an entanglement indicator, for which one would not have to know or to reconstruct the density matrix, which would not require lengthy optimization procedures, and which could be applied to any state. We present here such a method. We demonstrate that our method works for large classes of states of 2-qubit, 3-qubit, 4-qubit, and 5-qubit systems, and show that extension to larger systems is clearly achievable.

Our goals in this paper are: first, to show that our quantum neural network can learn an entanglement indicator that compares well with existing entanglement witnesses for small systems; and, second, to use neural network learning to construct a self-consistent entanglement indicator for larger systems for which entanglement is not yet well defined. That is, the information contained within the quantum system includes "the entanglement"; if we can successfully map that information onto an observable, we can extrapolate into regions of phase space which are not yet understood, and, thus, learn about entanglement in large systems. In the neural network literature this is called "bootstrapping"[8]: knowledge of part of a pattern allowing tentative inference of the rest.

2 Dynamic learning: quantum neural network (QNN)

We begin with the Schrödinger equation:

$$\frac{d\rho}{dt} = \frac{1}{i\hbar}[H, \rho] \tag{1}$$

where ρ is the density matrix and H is the Hamiltonian, whose formal solution[14] is

$$\rho(t) = \exp(iLt)\rho(0), \tag{2}$$

where L is the Liouville operator. We consider an N-qubit system whose Hamiltonian is:

$$H = \sum_{\alpha=1}^{N} K_\alpha \sigma_{x\alpha} + \varepsilon_\alpha \sigma_{z\alpha} + \sum_{\alpha\neq\beta=1}^{N} \zeta_{\alpha\beta}\sigma_{z\alpha}\sigma_{z\beta} \tag{3}$$

where $\{\sigma\}$ are the Pauli operators corresponding to each of the qubits, $\{K\}$ are the tunneling amplitudes, $\{\varepsilon\}$ are the biases, and $\{\zeta\}$, the qubit-qubit couplings. We choose the usual "charge basis ", in which each qubit's state is given as 0 or 1; for a system of N qubits there are 2^N states, each labelled by a bit string each of whose numbers corresponds to the state of each qubit, in order. The amplitude for each qubit to tunnel to its opposing state (i.e., switch between the 0 and 1 states) is its K value; each qubit has an external bias represented by its ε value; and each qubit is coupled to each of the other qubits, with a strength represented by

the appropriate ζ value. Note that, for example, the operator $\sigma_{xA} = \sigma_x \otimes I \ldots \otimes I$, where there are (N-1) outer products, acts nontrivially only on qubit A.

The parameters $\{K, \varepsilon, \zeta\}$ direct the time evolution of the system in the sense that, if one or more of them is changed, the way a given state will evolve in time will also change, because of Eqs. 1-3. This is the basis for using our quantum system as a neural network. There is a mathematical isomorphism between Eq. 2 and the equation for information propagation in a neural network: $\phi_{output} = F_W \phi_{input}$, where ϕ_{output} is the output vector of the network, ϕ_{input} the input vector, and F_W the network operator, which depends on the neuron connectivity weight matrix W. Here the role of the input vector is played by the initial density matrix $\rho(0)$, the role of the output by the density matrix at the final time, $\rho(t_f)$, and the role of the "weights" of the network by the parameters of the Hamiltonian, $\{K, \varepsilon, \zeta\}$, all of which can be adjusted experimentally[15]. By adjusting the parameters using a neural network type learning algorithm we can train the system to evolve in time from an input state to a set of particular final states at the final time t_f. Because the time evolution is quantum mechanical (and, we assume, coherent), a quantum mechanical function, like an entanglement witness of the initial state, can be mapped to an observable of the system's final state, a measurement made at the final time t_f. Complete details, including a derivation of the quantum dynamic learning paradigm using quantum backpropagation[16] in time[17], are given in [18].

The time evolution of the quantum system is calculated by integrating the Schrödinger equation numerically in MATLAB Simulink, using ODE4 (Runge-Kutta), with a fixed integration step size of 0.05 ns[19]. (Numerical diagonalization of the Hamiltonian gives the same results.) The system was initialized (prepared in) each input state, in turn, then allowed to evolve for 300 ns. A measurement is then made at the final time. All of the parameters $\{K, \varepsilon, \zeta\}$ were taken to be (piecewise constant) functions of time; this was done in the simulation by allowing them to change to a different constant value every Δt =75 ns (*i.e.*, four "time chunks".) Discretization error for the numerical integration was checked by redoing the calculations with a timestep of a tenth the size; results were not affected. For the backpropagation learning, the output error needs to be back-propagated through time[17], from t_f to 0. To implement this in MATLAB Simulink, a change of variable is made by letting $t' = t_f - t$, and running this simulation forward in t' in Simulink.

The picture is as follows: the system starts out in some state, whose entanglement we wish to know. We allow it to evolve in time, under a Hamiltonian which is piecewise constant in time, and has parameters which can be adjusted experimentally. At the final time we perform a measurement. This measurement is the "output" of the neural net, corresponding to the "input" of the initial state. We use neural network dynamic learning techniques to find that set of adjustable parameters for the Hamiltonian such that, for a (small) training set of initial states, the output numbers correspond well to the (known) entanglement of those initial states. We then use that (trained) set of parameters, first, to find if results on states not in the training set are also correct, and then, to extrapolate to larger classes of states. For reasons explored more completely in [18], we choose as our "output" for pairwise (two-qubit) entanglement, the square of the qubit-qubit correlation function $\langle \sigma_{zA}(t_f) \sigma_{zB}(t_f) \rangle^2$; the natural generalization, as the "output" for three-way entanglement, is (square of the) the three-point correlation function $\langle \sigma_{zA}(t_f) \sigma_{zB}(t_f) \sigma_{zC}(t_f) \rangle^2$; for four-way entanglement the four point correlation function, and so on.

There are, of course, other approaches we could use[20]. Our indicator for the pairwise entanglement of qubits A and B can also be written (for a pure state $|\psi\rangle$), $\langle\psi|U^\dagger\sigma_{zA}\sigma_{zB}U|\psi\rangle^2$, where the unitary matrix U is given by the product of the four (timechunk) propagators $e^{-iH\Delta t/\hbar}$. Instead of optimizing the parameters of the Hamiltonian, we could optimize the elements of U. However, our current approach makes the experimental implementation immediate; it is also an easier calculation, as follows. For n qubits, there are $2^{2n}-1$ such elements (taking, wolog, the determinant of U to be unity); our method requires determination of only $4[2n + n(n-1)/2]$ parameters; this is significantly smaller. In addition, our approach gives us a natural starting place for generalization from n to $n+1$ qubits, as we will see.

It is important to state here that we do not present a formula or algorithm for entanglement calculation. Instead, an entanglement indicator or witness emerges as a result of the learning process, that starts with data from known entanglement measures for the two-qubit system, and continues to three-, four-, and five-qubit systems. Indeed, since the entanglement is not, in fact, the expectation value of any operator, it is not possible to find a set of parameters that perfectly maps the entanglement onto the measure we have chosen, or any measure. For example, because of the phase oscillation problem[18, 21] we here consider only states with real coefficients, for which the entanglement is much better behaved; even so, our results are not perfect. Nonetheless an experimental indicator for which only a single measurement need be taken, for which one need not know or reconstruct the density matrix, and which does not require "closeness" to any one particular state, is, we think, of value. And while our method has limitations it also seems to be fairly easily generalizable to large systems for which analytical results are not available. This allows us, in a sense, to extract information about the entanglement from the system itself.

3 Why this is not quantum control

More generally, a nonlinear system can be written as

$$\dot{x} = f(x, u, p) \tag{4}$$

$$y = g(x, u) \tag{5}$$

where x is the state, f is a nonlinear function, g is an output measure, u is the system external control input, and y is the system output. x, u, and y are all functions of time, and p is a set of parameters. Notice from Eq. 1 that our *quantum* system is of the form $\dot{x} = Ax$, where the (complex) operator A contains the parameters $p = \{K, \varepsilon, \zeta\}$. In general p can be a set of a priori fixed system parameters (*e.g.* the mass, spring constant, and damping coefficient, in a mass spring damper system.) u is a control input that is not fixed ahead of time, and can be calculated on the fly and varied as the system is running, *i.e.*, for every different run of the system, u can be recalculated based, usually, on the current state of the system $x(t)$ as it is going from $x(0)$ to $x(t_f)$. So u can be different for different initial conditions ($x(0)$ values). While the set of parameters p can be functions of time as the system runs, they are fixed at the same values every time the system is run going from any $x(0)$ to $x(t_f)$, once determined.

A control problem, then, would be to calculate u on the fly in order to make the system states x or output y behave in a certain way. An optimal control problem would be to choose u to minimize some cost function $J(x, u)$ or $J(y, u)$. Given initial conditions $x(0)$ and u, the

system can be integrated from $t = 0$ to $t = t_f$ so that

$$x(t_f) = F(x(0), u, p) \tag{6}$$
$$y(t_f) = g(x(t_f), u(t_f)) \tag{7}$$

where $F = \int_0^{t_f} f(x, u, p)$. Our quantum system, on the other hand, has no external input, so it is written

$$x(t_f) = Q(x(0), p) \tag{8}$$
$$y(t_f) = m(x(t_f)) \tag{9}$$

where now Q defines the quantum system and m is some output measurement made on the system at the final time. The quantum computation problem is to find values for the specific fixed parameters p such that the quantum system approximates a given function $\hat{s} = h(\hat{r})$ at a finite number of points, M, in the domain, that is, $y(t_f) = \hat{s}_i$ when $x(0) = \hat{r}_i$ $i = 1, M$.

This is not a control problem, but a problem of finding a finite dimensional (with dimension equal to the number of parameters in p) subspace approximation of the function h. Interpreted as a quantum neural network, the weights p are found through a learning algorithm, where $[\hat{r}_i, \hat{s}_i]$, $i = 1, M$ are the training pairs in a training set, and the cost function to be minimized is $\hat{J} = [\hat{s}_i - y_i(t_f)]^2$ for each and every $i = 1, M$, where from Eqs. 8-9, $y_i(t_f)$=m(Q(x_i(0),p)) is the output for the system started in initial condition $x_i(0)$=r_i.

Our method could also be called quantum system design through learning[18], as machine learning is used to design an experimental quantum system to schieve a desired operational result.

4 Entanglement

For pure states, almost all the information contained within a system of 3 qubits is contained within the 2-particle reduced density matrices[9]. Thus it makes sense to train our entanglement indicator, to begin with, to the set of pairwise entanglements. Fortunately in previous work[18] we have already found a set of parameters that successfully maps the input state of a two-qubit system to a good approximation of the entanglement of formation, via the qubit-qubit correlation function at the final time, $\langle \sigma_{zA}(t_f)\sigma_{zB}(t_f)\rangle^2$. The parameters were found by training with a set of just four initial quantum states ("inputs"), as shown in Table 1: a fully entangled state ("Bell"), a "Flat" state (equal amounts of all basis states), a product state"Corr" whose initial ($t = 0$) correlation function $\langle \sigma_{zA}\sigma_{zB}\rangle^2$ is nonzero, and a partially entangled state "P". (Note here, again, that all coefficients are real.)

For a three-qubit system ABC there are three possible pairs: AB, AC, and BC; and therefore three different pairwise entanglements to train. We therefore used[22] a set of twelve training pairs: the four in Table 1 for each pair, outerproducted with $|0\rangle$ for the unpaired qubit. So, for example, the $Bell_{AC}$ state was taken to be $\frac{1}{\sqrt{2}}(|000\rangle + |101\rangle)$, and was trained to give an output of 1 for the network output measure $O_{AC} = \langle \sigma_{zA}(t_f)\sigma_{zC}(t_f)\rangle^2$ and zero for each of the output measures $O_{AB} = \langle \sigma_{zA}(t_f)\sigma_{zB}(t_f)\rangle^2$ and $O_{BC} = \langle \sigma_{zB}(t_f)\sigma_{zC}(t_f)\rangle^2$. The top two rows of Table 2 show the target entanglement indicator values ("Targets: Stage 2") ; the next three rows ("Trained") show the average calculated values after training to these twelve training pairs. Note that the third of the "Trained" rows (ABC) shows testing

on the three-way entanglement indicator $\langle\sigma_{zA}(t_f)\sigma_{zB}(t_f)\sigma_{zC}(t_f)\rangle^2$, which was, for Stage 2, not trained for; even so, its calculated value for $O_{ABC} = \langle\sigma_{zA}(t_f)\sigma_{zB}(t_f)\sigma_{zC}(t_f)\rangle^2$ is 0.57; this pattern can be seen again and again in successive training stages. We report here the averages; e.g., the $Bell_{\alpha\beta}$ column numbers are the averages over $\alpha\beta$ = AB, AC, and BC. ("Non-$\alpha\beta$" refers to the other two outputs (entanglement indicators) for each pair; there are six of these numbers.) More detailed procedures and results are in [22].

After the pairwise training was accomplished for the 3-qubit system, we then trained the residual[10] entanglement, as well. We did this by expanding the training set: adding one additional training pair, the three-way entangled Greenberger-Horne-Zeilinger (GHZ)[3] state for three qubits, $\frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$, for which we specified targets of zero for all three pairwise correlation functions, and 1 for an additional output measure (three-way entanglement indicator), the three-point correlation function $O_{ABC} = \langle\sigma_{zA}(t_f)\sigma_{zB}(t_f)\sigma_{zC}(t_f)\rangle^2$. The bottom rows in Table 2, (labelled "Targets: Stage 3" and "Trained")) display these targets and calculated average outputs.

Once trained, the parameters that are found can be tested, by using the Hamiltonian so defined to calculate our indicators for other initial states. (This will tell us whether we have successfully generalized our knowledge of entanglement, or have just been curvefitting the proverbial elephant.) Testing was therefore done on a large number of states not represented in the training set, including fully entangled states, partially entangled states, product (unentangled) states, and even mixed states. (Note that only pure states were present in the training sets.) Detailed results were presented for the three-qubit system in [22]; it is clear from the output matrix that there is definite separation, in two senses: first, that it is easy to see where the pairwise entanglement is (e.g., to distinguish between a state with AB entanglement and one with BC entanglement); and, second, that it is easy to differentiate among unentangled, partially entangled, and fully entangled states.

It was natural to take this one step further, to a system of four qubits. Again, we started from the already trained parameters for the smaller (3-qubit) system, and simply copied the additional parameters necessary (one tunneling parameter, one bias parameter, and three coupling parameters) from the trained parameters already found. Results are shown in Tables 3, 4, and 5. Again, we worked in stages: starting with a training set of 24 for the pairwise ($\alpha\beta$) training (6 pairwise Bell states, 6 pairwise flat states, 6 pairwise correlated unentangled states, and 6 pairwise partly entangled). Once those were trained, we added four more pairs corresponding to the four distinct 3-way GHZ states $\alpha\beta\gamma$; after those were trained, we added also the single 4-way GHZ state ABCD. Testing was then done, as with the 3-qubit system, on

Table 1. Input matrix for training the pairwise quantum neural network (QNN) entanglement witnesses. Each column is an input state, showing coefficients of the basis states (before normalization.) The N-qubit system was trained on this set of four for each of the $\binom{N}{2}$ pairs; this constituted the "pairwise" training.

Input	Bell	Flat	Corr.	P
$\|00\rangle$	1	1	0	1
$\|01\rangle$	0	1	0	1
$\|10\rangle$	0	1	0.5	1
$\|11\rangle$	1	1	1	0

Table 2. Stages 2 & 3: Three-qubit QNN entanglement targets and calculated average outputs. Each column is an input state corresponding to the respective column in Table 1; each row corresponds to a target function for the specified entanglement function. $\alpha\beta$ runs over all three pairs of qubits (AB, AC, and BC); ABC is the three-way entanglement. Stage 2 started from the trained 2-qubit system's parameters (stage 1)[18], and trained the three-qubit system to the set of pairwise entanglement training pairs(12 training pairs, 3 outputs per pair); Stage 3 used the pairwise set plus the GHZ state (13 training pairs, 4 outputs per pair). The double vertical line separates the states included in the training set from those only tested on. Nonzero target-training pairs are boldfaced for easy comparison; dashes indicate non-applicable table entries (*e.g.*, target values for states not trained at a given stage, or $\alpha\beta$ for a non-pair state.) RMS errors were: 1.2×10^{-3} for stage 2, and 1.8×10^{-3} for stage 3.

	O	$Bell_{\alpha\beta}$	Flat	Corr.	P	GHZ_3
Targets:	$\alpha\beta$	**1**	0	0	**0.44**	-
Stage 2	non-$\alpha\beta$	0	0	0	0	-
Trained	$\alpha\beta$	**0.9939**	0.0001	0.0004	**0.4392**	-
	non-$\alpha\beta$	0.0006	0.0006	0.0014	0.0006	0.0032
	ABC	0.0081	0.0001	0.0001	0.0037	0.5735
Targets:	$\alpha\beta$	**1**	0	0	**0.44**	-
Stage 3	non-$\alpha\beta$	0	0	0	0	0
	ABC	0	0	0	0	**1**
Trained	$\alpha\beta$	**0.9948**	0.0001	0.0009	**0.4389**	-
	non-$\alpha\beta$	0.0005	0.0012	0.0004	0.0022	0.0035
	ABC	0.0013	0.0006	0.0001	0.0012	**0.9883**

Table 3. Stage 4: Four-qubit QNN entanglement targets and calculated average outputs. $\alpha\beta$ runs over all six pairs of qubits; $\alpha\beta\gamma$ runs over all four triples of qubits; ABCD is the four-way entanglement. Stage 4 trained the four-qubit system to the set of 24 pairwise entanglement training pairs. Nonzero target-training pairs are boldfaced for easy comparison; dashes indicate non-applicable table entries (*e.g.*, target values for states not trained at a given stage, or $\alpha\beta$ for a non-pair state.) RMS error = 2.2×10^{-3} (24 training pairs, 6 outputs per pair).

	O	$Bell_{\alpha\beta}$	Flat	Corr.	P	$GHZ_{\alpha\beta\gamma}$	GHZ_4
Targets:	$\alpha\beta$	**1**	0	0	**0.44**	-	-
Stage 4	non-$\alpha\beta$	0	0	0	0	-	-
Trained	$\alpha\beta$	**0.9830**	0.0002	0.0008	**0.4426**	-	-
	non-$\alpha\beta$	0.0009	0.0024	0.0028	0.0017	0.0023	0.0032
	$\alpha\beta\gamma$	-	-	-	-	0.8566	-
	non-$\alpha\beta\gamma$	0.0007	0.0014	0.0012	0.0003	0.0001	0.0004
	ABCD	0.0021	0.0010	0.0000	0.0037	0.0020	0.9089

a similar (but expanded) set of states. Errors were slightly larger but the results maintained good separation. Buoyed by confidence in our method we then successfully expanded to a 5-qubit system, using an exactly similar procedure. Training results are shown in Tables 6, 7 and 8. Final values for the parameters are in Table 9.

5 Results and discussion

Training of the three-qubit system required considerable time: 5000 epochs (passes through the entire training set) to get pairwise entanglement from the trained results for the two-qubit system[22]. But training of the four-qubit system was (unexpectedly) easy. The training sets

Table 4. Stage 5: four-qubit QNN entanglement targets and calculated average outputs. Stage 5 added the four 3-way GHZ states to the training set of Table 3. Nonzero target-training pairs are boldfaced for easy comparison; dashes indicate non-applicable table entries (*e.g.*, target values for states not trained at a given stage, or $\alpha\beta$ for a non-pair state.) RMS error = 1.9×10^{-3} (28 training pairs, 10 outputs per pair).

	O	$Bell_{\alpha\beta}$	Flat	Corr.	P	$GHZ_{\alpha\beta\gamma}$	GHZ_4
Targets:	$\alpha\beta$	**1**	0	0	**0.44**	-	-
Stage 5	non-$\alpha\beta$	0	0	0	0	0	-
	$\alpha\beta\gamma$	-	-	-	-	**1**	-
	non-$\alpha\beta\gamma$	0	0	0	0	0	-
Trained	$\alpha\beta$	**0.9731**	0.0001	0.0005	**0.4360**	-	-
	non-$\alpha\beta$	0.0006	0.0023	0.0024	0.0019	0.0023	0.0038
	$\alpha\beta\gamma$	-	-	-	-	**0.9728**	-
	non-$\alpha\beta\gamma$	0.0003	0.0014	0.0012	0.0003	0.0001	0.0007
	ABCD	0.0014	0.0012	0.0000	0.0036	0.0010	0.7649

Table 5. Stage 6: four-qubit QNN entanglement targets and calculated average outputs. Stage 6 added the one 4-way GHZ state to the training set of stage 5. Nonzero target-training pairs are boldfaced for easy comparison; dashes indicate non-applicable table entries (*e.g.*, target values for states not trained at a given stage, or $\alpha\beta$ for a non-pair state.) RMS error = 2.2×10^{-3} (29 training pairs, 11 outputs per pair).

	O	$Bell_{\alpha\beta}$	Flat	Corr.	P	$GHZ_{\alpha\beta\gamma}$	GHZ_4
Targets:	$\alpha\beta$	**1**	0	0	**0.44**	-	-
Stage 6	non-$\alpha\beta$	0	0	0	0	0	0
	$\alpha\beta\gamma$	-	-	-	-	**1**	-
	non-$\alpha\beta\gamma$	0	0	0	0	0	0
	ABCD	0	0	0	0	0	**1**
Trained	$\alpha\beta$	**0.9682**	0.0000	0.0002	**0.4341**	-	-
	non-$\alpha\beta$	0.0002	0.0009	0.0011	0.0007	0.0010	0.0017
	$\alpha\beta\gamma$	-	-	-	-	**0.9592**	-
	non-$\alpha\beta\gamma$	0.0002	0.0005	0.0004	0.0002	0.0001	0.0003
	ABCD	0.0004	0.0003	0.0000	0.0011	0.0014	**0.9488**

were, of course, larger (training set size is noted in each table caption): a 3-qubit system has only three distinct pairs (and, therefore, pairwise entanglements), and only one triplet, while the number of ways for the 4-qubit system are 6 and 4, respectively, with one quad (and so on, following the binomial coefficients.) However the 4-qubit system required much less training: only 100 epochs at each stage. The 5-qubit system was also trained for 100 epochs at each stage, but didn't really need so much: 98% of the change in the parameters, as well as in the error reduction, had been accomplished by less than half that time, as shown in Figure 1. Successively larger systems still need to be trained to decrease initial errors; yet the amount of training necessary seems to diminish with successive training stages.

Figures 2 and 3 make this point graphically. They show the evolution of the tunneling, bias, and coupling parameters, from initial training of the 2-qubit system, through each stage of the training of the 3-qubit, 4-qubit, and 5-qubit systems. Initially the QNN's learning entanglement requires large changes in the parameters, but as the size and complexity of the

Table 6. Stages 7 & 8: five-qubit QNN entanglement targets and calculated average outputs. $\alpha\beta$ runs over all ten pairs of qubits; $\alpha\beta\gamma$ runs over all ten triples of qubits; $\alpha\beta\gamma\delta$ runs over all five quadruples of qubits; ABCDE is the five-way entanglement. Stage 7 trained the five-qubit system to the set of 40 pairwise entanglement states for the 5-qubit system; Stage 8 added the ten 3-way GHZ states. Nonzero target-training pairs are boldfaced for easy comparison; dashes indicate non-applicable table entries (*e.g.*, target values for states not trained at a given stage, or $\alpha\beta$ for a non-pair state.) RMS errors were: 5.6×10^{-4} for stage 7 (40 training pairs, 10 outputs per pair), and 7.9×10^{-4} for stage 8 (50 training pairs, 20 outputs per pair).

	O	$Bell_{\alpha\beta}$	Flat	Corr.	P	$GHZ_{\alpha\beta\gamma}$	$GHZ_{\alpha\beta\gamma\delta}$	GHZ_5
Targets:	$\alpha\beta$	**1**	0	0	**0.44**	-	-	-
Stage 7	non-$\alpha\beta$	0	0	0	0	-	-	-
Trained	$\alpha\beta$	**0.9908**	0.0000	0.0003	**0.4400**	-	-	-
	non-$\alpha\beta$	0.0003	0.0005	0.0010	0.0003	0.0003	0.0005	0.0007
	$\alpha\beta\gamma$	-	-	-	-	0.7055	-	-
	non-$\alpha\beta\gamma$	0.0002	0.0003	0.0003	0.0001	0.0001	0.0001	0.0022
	$\alpha\beta\gamma\delta$	-	-	-	-	-	0.7806	-
	non-$\alpha\beta\gamma\delta$	0.0004	0.0001	0.0000	0.0003	0.0015	0.0001	0.0013
	ABCDE	0.0000	0.0000	0.0000	0.0000	0.0005	0.0039	0.8307
Targets:	$\alpha\beta$	**1**	0	0	**0.44**	-	-	-
Stage 8	non-$\alpha\beta$	0	0	0	0	0	-	-
	$\alpha\beta\gamma$	-	-	-	-	**1**	-	-
	non-$\alpha\beta\gamma$	0	0	0	0	0	-	-
Trained	$\alpha\beta$	**0.9754**	0.0000	0.0002	**0.4323**	-	-	-
	non-$\alpha\beta$	0.0001	0.0005	0.0005	0.0005	0.0003	0.0003	0.0001
	$\alpha\beta\gamma$	-	-	-	-	**0.9765**	-	-
	non-$\alpha\beta\gamma$	0.0001	0.0002	0.0001	0.0001	0.0001	0.0006	0.0006
	$\alpha\beta\gamma\delta$	-	-	-	-	-	0.7597	-
	non-$\alpha\beta\gamma\delta$	0.0002	0.0000	0.0000	0.0002	0.0004	0.0002	0.0085
	ABCDE	0.0000	0.0000	0.0000	0.0000	0.0000	0.0090	0.0612

system is increased, the relative size of the changes decreases, till by stage 10 (the 5-qubit system's training to the full set of 56 training pairs), changes in the parameters are just a few percent. Note that in the two figures, the relative sizes of the final numbers (root square percent change in K_a is 3.6%; in ζ_{AB}, 6.7%) are comparable. This is reminiscent of the well-known neural network technique called "bootstrapping"[8]: in a sense, information about a smaller system can, in part, generalize to a larger. This makes us hopeful that this technique may be viable for use even on large systems.

Figure 4 shows the average total pairwise entanglement (sum over all outputs) of the W_k states as a function of k, for k = 1 to 5, as computed by the QNN using the trained 5-qubit system. This is our computational analog of the measurements done by Kimble et al.[23] of the number of modes (k) sharing a single excitation photon. Our results show similar separations among states of different k. These numbers change only very slightly when we look at so-called "flipped" W states[11] instead (where, instead of a single excitation shared among N qubits, we have a single 0 so shared - e.g., $\tilde{W}_3 = \frac{1}{\sqrt{3}}(|110\rangle \pm |101\rangle \pm |011\rangle)$).

Figure 5 shows the pairwise entanglements of the (normalized) states $\alpha|0001\rangle + \beta|0010\rangle + |0100\rangle + |1000\rangle$, calculated by the QNN using the trained 4-qubit system, as a function of

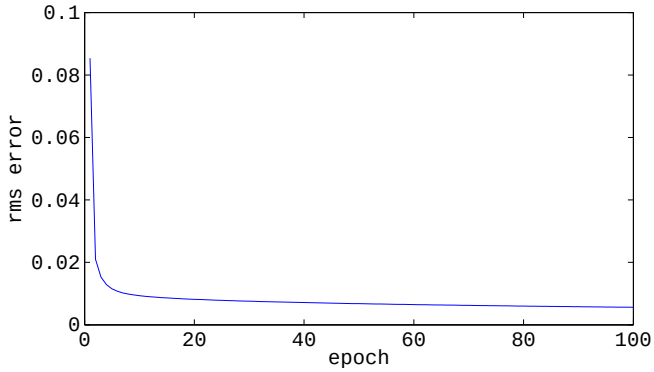

Fig. 1. Root mean squared error per training pair as a function of epoch (pass through the training set), for the 5-qubit system on the pairwise training set (stage 7), summed over the 10 outputs per pair.

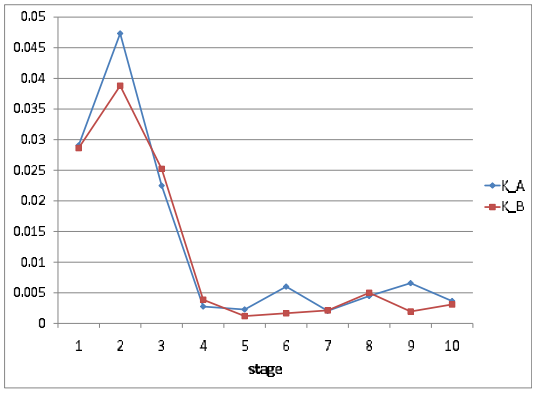

Fig. 2. Root square relative difference, $\sqrt{\sum \Delta K^2}/\sum K$, for both K_A and K_B, as a function of training stage. The sums are over the timeslices, and the differences are to each successive stage. Stage 1 is 2-qubit training[18]; stage 2, 3-qubit pairwise; stage 3, 3q pairwise plus 3-way; stage 4, 4q pairwise; stage 5, 4q pairwise plus 3-way; stage 6, 4q pairwise plus 3-way plus 4-way; stage 7, 5q pairwise; stage 8, 5q pairwise plus 3-way; stage 9, 5q pairwise plus 3-way plus 4-way; stage 10, 5q pairwise plus 3-way plus 4-way plus 5-way.

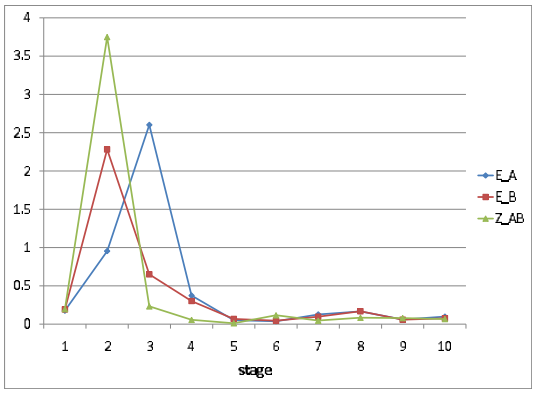

Fig. 3. Root square relative difference in each of the parameters ε_A, ε_B and ζ_{AB} as functions of training stage. Stages as in Figure 2.

Table 7. Stage 9: five qubit QNN entanglement targets and calculated average outputs. Stage 9 added the five 4-way GHZ states to the training set of stage 8. Nonzero target-training pairs are boldfaced for easy comparison; dashes indicate non-applicable table entries (*e.g.*, target values for states not trained at a given stage, or $\alpha\beta$ for a non-pair state.) RMS error = 1.4×10^{-3} (55 training pairs, 25 outputs per pair).

	O	$Bell_{\alpha\beta}$	Flat	Corr.	P	$GHZ_{\alpha\beta\gamma}$	$GHZ_{\alpha\beta\gamma\delta}$	GHZ_5
Targets:	$\alpha\beta$	**1**	0	0	**0.44**	-	-	-
Stage 9	non-$\alpha\beta$	0	0	0	0	0	0	-
	$\alpha\beta\gamma$	-	-	-	-	**1**	-	-
	non-$\alpha\beta\gamma$	0	0	0	0	0	0	-
	$\alpha\beta\gamma\delta$	-	-	-	-	-	**1**	-
	non-$\alpha\beta\gamma\delta$	0	0	0	0	0	0	-
Trained	$\alpha\beta$	**0.9647**	0.0000	0.0001	**0.4315**	-	-	-
	non-$\alpha\beta$	0.0000	0.0002	0.0002	0.0002	0.0001	0.0001	0.0002
	$\alpha\beta\gamma$	-	-	-	-	**0.9574**	-	-
	non-$\alpha\beta\gamma$	0.0000	0.0001	0.0000	0.0001	0.0000	0.0002	0.0005
	$\alpha\beta\gamma\delta$	-	-	-	-	-	**0.9630**	-
	non-$\alpha\beta\gamma\delta$	0.0001	0.0000	0.0000	0.0000	0.0002	0.0001	0.0015
	ABCDE	0.0000	0.0000	0.0000	0.0000	0.0002	0.0022	0.1855

Table 8. Stage 10: five-qubit QNN entanglement targets and calculated average outputs. Stage 10 added the one 5-way GHZ state to the training set of stage 9. Nonzero target-training pairs are boldfaced for easy comparison; dashes indicate non-applicable table entries (*e.g.*, target values for states not trained at a given stage, or $\alpha\beta$ for a non-pair state.) RMS error = 2.4×10^{-3} (56 training pairs, 26 outputs per pair).

	O	$Bell_{\alpha\beta}$	Flat	Corr.	P	$GHZ_{\alpha\beta\gamma}$	$GHZ_{\alpha\beta\gamma\delta}$	GHZ_5
Targets:	$\alpha\beta$	**1**	0	0	**0.44**	-	-	-
Stage 10	non-$\alpha\beta$	0	0	0	0	0	0	0
	$\alpha\beta\gamma$	-	-	-	-	**1**	-	-
	non-$\alpha\beta\gamma$	0	0	0	0	0	0	0
	$\alpha\beta\gamma\delta$	-	-	-	-	-	**1**	-
	non-$\alpha\beta\gamma\delta$	0	0	0	0	0	0	0
	ABCDE	0	0	0	0	0	0	**1**
Trained	$\alpha\beta$	**0.9148**	0.0000	0.0000	**0.4295**	-	-	-
	non-$\alpha\beta$	0.0000	0.0001	0.0001	0.0001	0.0001	0.0000	0.0003
	$\alpha\beta\gamma$	-	-	-	-	**0.9195**	-	-
	non-$\alpha\beta\gamma$	0.0000	0.0005	0.0004	0.0002	0.0000	0.0000	0.0005
	$\alpha\beta\gamma\delta$	-	-	-	-	-	**0.9195**	-
	non-$\alpha\beta\gamma\delta$	0.0000	0.0000	0.0000	0.0000	0.0001	0.0000	0.0004
	ABCDE	0.0000	0.0000	0.0000	0.0000	0.0001	0.0001	**0.8258**

both α and β. Results are not significantly different when using the trained 5-qubit system. When $\alpha = 0$ and $\beta = 0$, this is $(|01\rangle + |10\rangle) \otimes |00\rangle = EPR_{AB} \otimes |00\rangle$, which has full pairwise AB entanglement while all other entanglements are zero; our results match this. When $\alpha = 1$ and $\beta = 0$, this is $|0001\rangle + |0100\rangle + |1000\rangle$, which is the three-way W state in qubits ABD. In terms of pairwise entanglement, a three-way W state is equally pairwise entangled in all three possible combinations AB, AD, and BD, so we would expect the red, green, and blue surfaces

Table 9. Final (stage 10) trained parameters for entanglement of 5-qubit system, in MHz. Each row is a different parameter, as labelled. Each parameter is a function of time; the first row shows the timeslice number.

t	1	2	3	4
K_A	2.1733	2.4021	2.4688	2.4139
K_B	2.5842	2.1591	2.2702	2.4710
K_C	2.3984	2.3252	2.3501	2.4294
K_D	2.3893	2.3128	2.3451	2.4327
K_E	2.3926	2.3144	2.3446	2.4274
ε_A	0.7443	-0.1558	-1.1248	-0.7230
ε_B	0.9903	-0.4483	-1.0216	-0.4294
ε_C	1.1027	-0.4998	-0.8678	-0.4752
ε_D	0.7783	-0.2453	-1.1015	-0.6125
ε_E	0.7703	-0.2472	-1.0941	-0.6116
ζ_{AB}	-0.3576	0.2797	-0.5974	0.1521
ζ_{AC}	-0.1761	0.1800	-0.5796	-0.0604
ζ_{AD}	-0.1687	0.1802	-0.5558	-0.0471
ζ_{AE}	-0.1563	0.1758	-0.5324	-0.0402
ζ_{BC}	-0.2228	0.1806	-0.5462	0.0340
ζ_{BD}	-0.2138	0.1808	-0.5227	0.0431
ζ_{BE}	-0.2026	0.1785	-0.5014	0.0484
ζ_{CD}	-0.3052	0.2375	-0.4795	0.1959
ζ_{CE}	-0.2965	0.2358	-0.4638	0.1801
ζ_{DE}	-0.2896	0.2265	-0.4465	0.1771

to come together at that point, while the yellow (AC), cyan (BC) and magenta (CD) surfaces are zero there. This is in fact what we see in the figure. Similarly, when $\alpha = 0$ and $\beta = 1$, this is $|0010\rangle + |0100\rangle + |1000\rangle$, which is the three-way W state in qubits ABC; at this point the AB (red), AC (yellow), and BC (cyan) entanglements are all approximately equal and the AD (green), BD (blue), and CD (magenta) entanglements are zero. When both $\alpha = 1$ and $\beta = 1$, this is the four-way W state $|001\rangle + |0010\rangle + |0100\rangle + |1000\rangle$, equally entangled in all six pairwise ways AB (red), AC (yellow), AD (green), BC (cyan), BD (blue), and CD (magenta).

The value for the pairwise entanglement of the three-way W states deserves some further comment. In the original training set for the two-qubit system, it was found[18] that the value for the entanglement of the "P" state, $|01\rangle + |10\rangle + |11\rangle$, that resulted in the smallest overall error (training plus testing), was 0.44: that is, using this value, rather than some other, led to the greatest possible self-consistency for the QNN method. We therefore used this same value for $P \otimes |0\rangle$ for each of the three pairs, in training the three-qubit system[22]. The three-way W state $W_{ABC} = |001\rangle + |010\rangle + |100\rangle$ is not a "P" state in any of the three pairs, of course, but it is related in the following sense: we can write $W_{ABC} = |001\rangle + (|01\rangle + |10\rangle) \otimes |0\rangle$, that is, an EPR state in two of the qubits, plus a contamination term. In the same way, we can think of "P" as being a (fully entangled) EPR state, plus an amount of contamination, the inclusion of which diminishes the entanglement. Thus for self-consistency we would expect each of the pairwise entanglements of a W_3 state to be approximately 0.44, as in fact occurs.

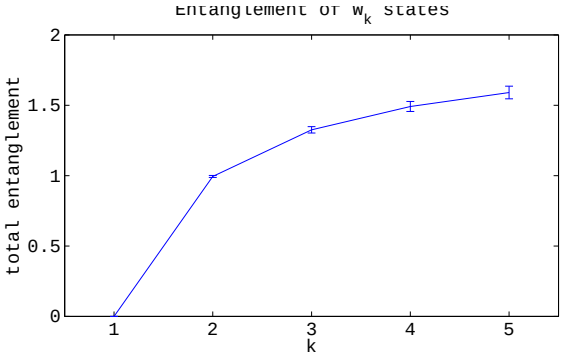

Fig. 4. Total (pairwise) entanglement of W_k, calculated by the QNN using the trained parameters listed in Table 9, as a function of k. Each data point is the average of the total entanglement for the $2^{k-1}\begin{pmatrix} 5 \\ k \end{pmatrix}$ states of the form $\mathrm{W}_k \otimes |0\rangle^{N-k}$. Error bars show the standard deviation at each point, which increases from from 0.7%, for the W_2 state, to 2.8% for the W_5 state. The line is drawn to guide the eye.

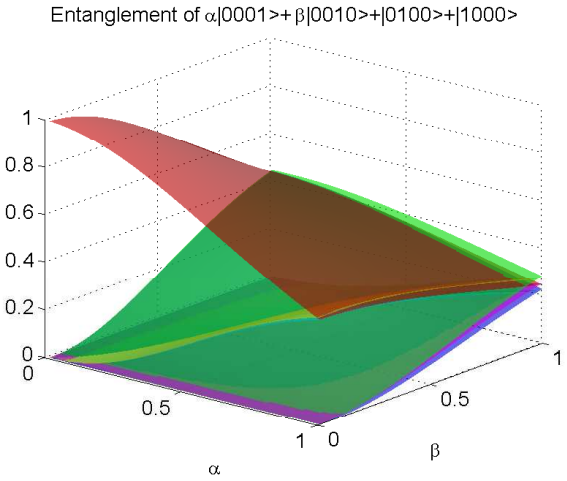

Fig. 5. Pairwise entanglement of $\alpha|0001\rangle + \beta|0010\rangle + |0100\rangle + |1000\rangle$, as a function of α and β, as calculated by the QNN. Each color represents a different entanglement measure: red, the pairwise entanglement between A and B, O_{AB}; yellow, that between A and C, O_{AC}; green, O_{AD}; cyan, O_{BC}; blue, O_{BD}; and magenta, O_{CD}.

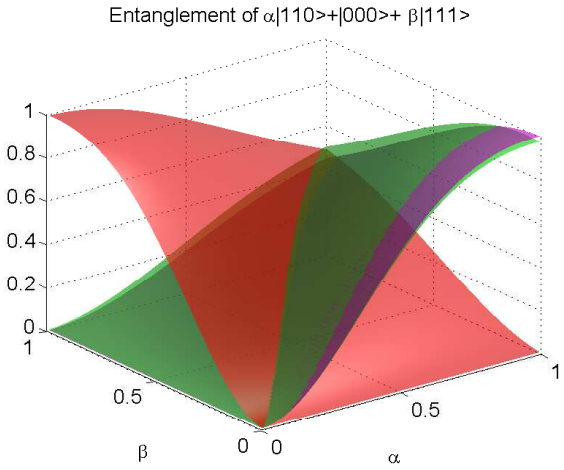

Fig. 6. Entanglement of $\alpha|110\rangle + \beta|111\rangle + |000\rangle$, as a function of α and β, as calculated by the QNN, and compared with the 3-tangle[10]. The red surface shows the pairwise entanglement between qubits A and B, O_{AB}; the green, three-way entanglement among A, B, and C,O_{ABC}; and magenta is the 3-tangle.

(It should be noted here that this is an internal self-consistency; we do not know whether the number itself has any physical meaning, and we claim none.)

We can also look at states which have three-way (or higher) entanglement. Figure 6 shows the pairwise (red) and three-way (green) entanglement of the states $\alpha|110\rangle + \beta|111\rangle + |000\rangle$, calculated by the QNN using the trained 3-qubit system, as a function of α and β, and compared with the residual or 3-tangle of Coffman[10] (magenta). Results were not significantly different when using the trained 4-qubit or five-qubit systems. When $\alpha = 0$ and $\beta = 1$, this is the pure GHZ state for three qubits, and the three-way entanglement is maximal; both the 3-tangle and the QNN calculate this to be one. Indeed the QNN three-way entanglement (green) tracks the 3-tangle (magenta) fairly well over the entire range of parameters shown. This can also be seen in Figure 7, where we show the tracking of both pairwise and residual tangle by the QNN for states of the form $|100\rangle + \beta|010\rangle + \gamma|001\rangle$.

With the trained net, we can evaluate any pairwise or N-wise entanglement for mixed states, as well, without having to find an optimal decomposition[11]. Figure 8 shows the computed entanglement of states of the form $\alpha|GHZ_5\rangle\langle GHZ_5| + \beta|W_5\rangle\langle W_5| + (1 - \alpha - \beta)|\tilde{W}_5\rangle\langle|\tilde{W}_5|$, as a function of both α and β. As expected from symmetry, all the pairwise entanglements lie atop one another, so only one is shown, for clarity; all the three-way and four-way entanglements are zero. As this is a mixture, not a superposition, there is no interference among the three contributing states, and the $|W_5\rangle$ and $|\tilde{W}_5\rangle$ states contribute equivalently to the pairwise (red) entanglements. Compare this relatively featureless behavior with that shown in Figure 9, which shows the calculated pairwise (red), four-way (blue), and five-way (green) entanglements of the superposition states $\alpha|GHZ_5\rangle + \beta|W_5\rangle + (1-\alpha-\beta)|\tilde{W}_5\rangle$. Again, all the pairwise entanglements (red) lie atop each other, by symmetry, as do the three-way (not shown) and the four-way (blue). The three-way entanglements are still all zero; however, the four-way (blue) are not, because there are combinations of the coefficients such

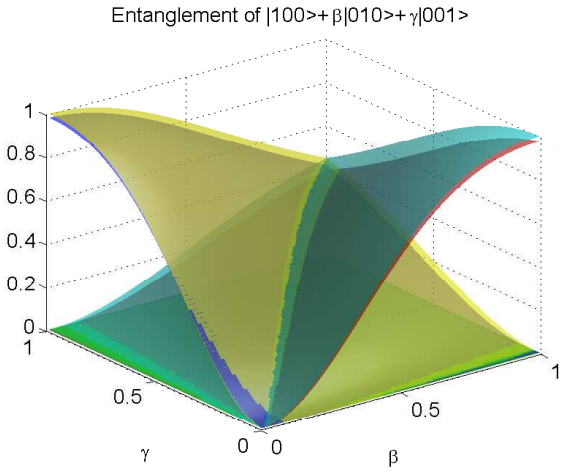

Fig. 7. Entanglement of $|100\rangle + \beta|010\rangle + \gamma|001\rangle$, as a function of γ and β, as calculated by the QNN, and compared with analytical results. Each color corresponds to a different entanglement measure: red, pairwise AB entanglement, O_{AB}; and blue, pairwise AC entanglement, O_{AC}, both computed by the QNN. Compare these to τ_{AB}, the tangle between A and B, in cyan; and τ_{AC}, the tangle between A and C, in yellow (both computed analytically.) The residual tangle is identically zero for all these states, in agreement with the QNN's calculated O_{ABC}, in green.

that the superposition contains some nonzero amplitude of each of the $|GHZ_4\rangle$ states; in neither $|W_5\rangle$ nor $|\tilde{W}_5\rangle$ are there any states with exactly three qubits in the same state, so there is no three-way entanglement. In both Figure 9 and Figure 8 we see the expected limiting behavior: for $\alpha = 1$ and $\beta = 0$, we have maximal five-way entanglement only; for $\alpha = 0$ and $\beta = 1$, we have maximal pairwise entanglement only, spread over all ten possible pairs (see also Figure 4.)

6 Conclusions

We have presented a new approach for pairwise and multiqubit entanglement, in which we use a quantum neural network to create an entanglement indicator by extrapolation from accepted particular cases. Our method is applicable to any state, whether pure or mixed. Using dynamic quantum backpropagation learning, we have shown that a quantum system can be trained, in a sense, to compute its own degree of entanglement. No prior state reconstruction or tedious optimization procedure is necessary, nor is "closeness" to any particular state. We envision an experimental implementation in which parameters could be, to begin with, roughly estimated from simulations such as those presented here (in the neural network literature, this is called "offline" or "batch" training[24].) These parameters could then be refined experimentally ("online" training of the neural network.) Because the state of the system at each timeslice must be known in order to do quantum backprop, it can only be done in simulation; a different parameter adjustment method would be used experimentally, *e.g.*, a reinforcement or genetic algorithm. The experimental refinement procedure might also provide for the inclusion of features and interactions present in the physical state but not in the (simplified) model. In any case, once good values for the parameters are determined, any state's entanglement can be approximately found by experimental measurement. And

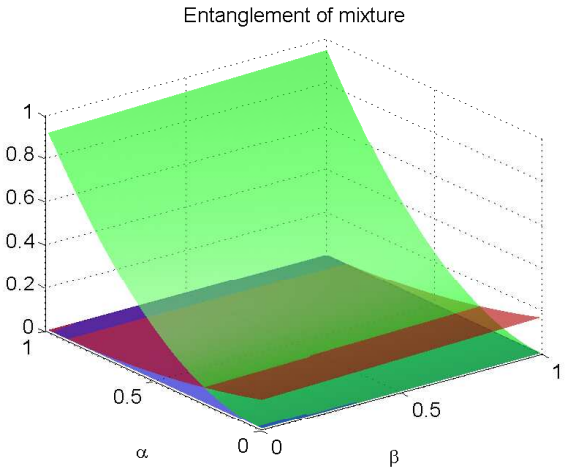

Fig. 8. Entanglement of the mixed states $\alpha|GHZ_5\rangle\langle GHZ_5| + \beta|W_5\rangle\langle W_5| + (1-\alpha-\beta)|\tilde{W}_5\rangle\langle|\tilde{W}_5|$, as a function of both α and β. Each color represents a different entanglement measure: red, the pairwise entanglement; blue, four-way entanglement; and green, five-way entanglement.

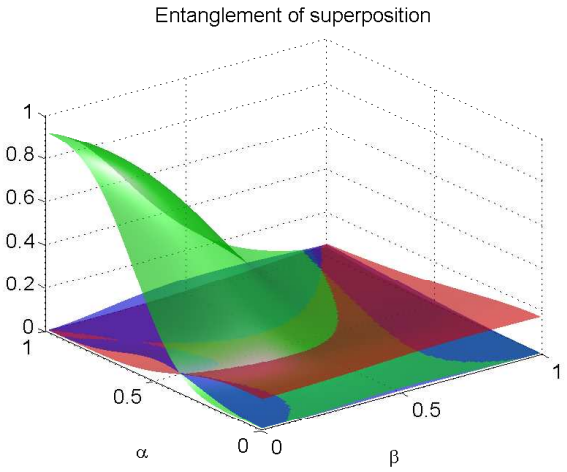

Fig. 9. Entanglement of the superposition states $\alpha|GHZ_5\rangle + \beta|W_5\rangle + (1-\alpha-\beta)|\tilde{W}_5\rangle$, as a function of both α and β. Each color represents a different entanglement measure: red, the pairwise entanglement; blue, four-way entanglement; and green, five-way entanglement.

while we have only carried out simulations through a system size of 5 qubits, the additional training necessary significantly decreases as the size of the system grows, raising hopes for the method's feasibility even for large qubit networks.

It should be emphasized, though, that this method provides only an indicator of the entanglement (a "witness", not a "measure.") Agreement with analytical results, while good, is not exact - nor, indeed, can it be, as discussed earlier. And while agreement is here demonstrated for some classes of states, the testing shown is not exhaustive even for states with real coefficients: large classes of states may be being missed. Since there is a natural measure on the set of real state vectors (the uniform measure on the unit sphere), the notion of an average agreement can be well defined, which will allow us to do systematic testing on large random sets. This work is in progress. More important, we have not explored here application to phase shifted superpositions and mixtures. This is a much harder problem to solve, either analytically or computationally, since entanglement depends on phase in complicated ways. Any single-measurement witness must[18] exhibit oscillation as a function of relative phase, but it is not at all clear what is the most efficient way of dealing with this[11, 12, 13, 25]. Quantum tomography obviates the problem but rapidly becomes impractical with increasing system size. It is possible that we will see an increased efficiency with size in the complex case, as we saw in with real coefficients. Further work is needed in all these areas, and is ongoing[21].

That being said, we believe our present work is still of great value, providing as it does a way of bypassing difficult analytical work in addressing the need[23] for experimental measurements to determine entanglement. Moreover, our approach may have other important advantages. Classically neural networks have proven fault tolerant and robust to noise; they are also famously used for noise reduction in signals. In quantum systems there is also the problem of decoherence. It is possible that quantum neural networks may be well suited for dealing with these types of problems in quantum computing. There are undoubtedly many other possible applications for the use of learning or AI methods as shortcuts to constructing quantum algorithms.

Acknowledgements

This work was supported in part by the National Science Foundation under Grant No. NSF PHY05-51164, through the KITP Scholars program (ECB), at the Kavli Institute for Theoretical Physics, University of California at Santa Barbara, Santa Barbara, CA. We thank W.K. Wootters, J.F. Behrman and J. Watkins for helpful discussions, and an anonymous referee for helpful suggestions.

References

1. C.H. Bennett, D.P. DiVincenzo, J.A. Smolin, and W.K. Wootters (1996), *Mixed-state entanglement and quantum error correction*, Phys. Rev. A 54, pp. 3824-3851.
2. W.K. Wootters (1998), *Entanglement of formation of an arbitrary state of two qubits* , Phys. Rev. Lett. 80, pp. 2245-2248.
3. D.M. Greenberger, M.A. Horne, and A. Zeilinger (1989), in *Bell's Theorem and the Conception of the Universe*, M. Kafatos, ed. , Kluwer Acdemic (Dordrecht), p 107.

4. V. Vedral, M.B. Plenio, M.A. Rippin, and P.L. Knight (1997), *Quantifying entanglement*, Phys. Rev. Lett. 78, pp. 2275-2279; V Vedral and M.B. Plenio (1998), *Entanglement measures and purification procedures*, Phys. Rev. A 57, pp. 1619-1633; L. Henderson and V. Vedral (2001), *Classical, quantum and total correlations*, J. Phys. A 34, pp. 6899-6905.
5. S. Tamaryan, A. Sudbery, and L. Tamaryan (2010), *Duality and the geometric measure of entanglement of general multiqubit W states*, Phys. Rev. A 81, 052319.
6. H.S. Park, S.-S.B. Lee, H. Kim, S.-K. Choi, and H.-S. Sim (2010), *Construction of optimal witness for unknown two-qubit entanglement*, Phys. Rev. Lett. 105, 230404.
7. R. Filip (2002), *Overlap and entanglement-witness measurements* , Phys. Rev. A 65, 062320; F.G.S.L. Brando (2005), *Quantifying entanglement with witness operators* , quant-ph/0503152.
8. B. Efron and R.J. Tibshirani (1994), *An Introduction to the bootstrap*. Boca Raton, FL: Chapman and Hall/CRC.
9. W.K. Wootters (2002), *The power of reduced quantum states*, Phys. Rev. Lett. 89, 207901; *High order correlations of generic pure states of finite-dimensional quantum systems are determined by lower order correlations*, Phys. Rev. Lett. 89, 277906.
10. V. Coffman, J. Kundu, and W.K. Wootters (2000), *Distributed entanglement*, Phys. Rev. A 61, 052306.
11. E. Jung, M.-R. Hwang, D.K. Park, and J.W. Son (2009), *Three-tangle for rank-3 mixed states: mixture of Greenberger-Horne-Zeilinger, W and flipped W states*, Phys. Rev. A **79**, 024306.
12. R. Lohmayer, A. Osterloh, J. Siewert, and A. Uhlmann (2006), *Entangled three-qubit states without concurrence and three-tangle*, Phys. Rev. Lett. 97, 260502.
13. Y.-K. Bai, M.-Y. Ye, and Z.D. Wang (2008), *Entanglement in a class of multiqubit mixed states without multipartite tangles*, Phys. Rev. A **78**, 062325.
14. A. Peres (1995), *Quantum Theory: Concepts and Methods*. Dordrecht, The Netherlands: Kluwer.
15. T. Yamamoto, Yu.A. Pashkin, O. Astafiev, Y. Nakamura, and J.S. Tsai (2003), *Demonstration of conditional gate operation using superconducting charge qubits*, Nature 425, pp. 941-944.
16. Yann le Cun (1988), *A theoretical framework for back-propagation* in *Proc. 1998 Connectionist Models Summer School,* D. Touretzky, G. Hinton, and T. Sejnowski, eds., Morgan Kaufmann, (San Mateo), pp. 21-28.
17. Paul Werbos (1992), in *Handbook of Intelligent Control* , Van Nostrand Reinhold, pp. 79-80 and 339-344.
18. E.C. Behrman, J.E. Steck, P. Kumar, and K.A. Walsh (2008), *Quantum algorithm design using dynamic learning*, Quantum Information and Computation 8, pp. 12-29.
19. MATLAB Simulink documentation notes [Online]. Available at http://www.mathworks.com/access/helpdesk/help/toolbox/simulink
20. We thank an anonymous referee for pointing this out.
21. R.E. Bonde, E.C. Behrman and J.E. Steck (2012), *Phase oscillation in entanglement witnesses*, in preparation.
22. E.C. Behrman and J.E. Steck (2011),"Dynamic learning of pairwise and three-way entanglement," in *Proceedings of the Third World Congress on Nature and Biologically Inspired Computing (NaBIC)*(IEEE: Salamanca, Spain, October 19-21, 2011); also at arXiv:quant-ph/1106.4254.
23. S. B. Papp, K. S. Choi, H. Deng, P. Lougovski, S. J. van Enk and H. J. Kimble (2009), *Characterization of Multipartite Entanglement for One Photon Shared Among Four Optical Modes*, Science 324, 764.
24. P.D. Wasserman (1993), *Advanced Methods in Neural Computing*. New York: Van Nostrand Reinhold.
25. C-P Yang and S. Han (2005), *Extracting an arbitrary relative phase from a multiqubit two-component entangled state*, Phys. Rev. A 72, 014306.