ECAI 2020
G.D. Giacomo et al. (Eds.)

doi:10.3233/FAIA200315

Finding Quantum Critical Points with Neural-Network Quantum States

Remmy Zen[1] and **Long My**[2] and **Ryan Tan**[3] and **Frédéric Hébert**[4] and **Mario Gattobigio**[4] and **Christian Miniatura**[5,2,6,7,8,4] and **Dario Poletti**[9,3,5] and **Stéphane Bressan**[1]

Abstract. Finding the precise location of quantum critical points is of particular importance to characterise quantum many-body systems at zero temperature. However, quantum many-body systems are notoriously hard to study because the dimension of their Hilbert space increases exponentially with their size. Recently, machine learning tools known as neural-network quantum states have been shown to effectively and efficiently simulate quantum many-body systems. We present an approach to finding the quantum critical points of the quantum Ising model using neural-network quantum states, analytically constructed innate restricted Boltzmann machines, transfer learning and unsupervised learning. We validate the approach and evaluate its efficiency and effectiveness in comparison with other traditional approaches.

1 INTRODUCTION

Quantum critical points [35] mark the transition between different phases of quantum many-body systems [33] at zero temperature. Finding their precise location is of particular importance to characterise the physical properties of quantum many-body systems [30]. However, these systems are notoriously hard to study because the dimension of the Hilbert space of their wave functions, being the tensor product of the constituents of the systems, increases exponentially with their size. From a practical point of view however, one is often interested in the low-energy sector of the system, which generally involves quantum states living in a much smaller portion of the full Hilbert space. In this case, these computational complexity issues can be alleviated: Both deterministic and stochastic approximation algorithms have been proposed to find this relevant portion of the Hilbert space and to accurately describe the system.

Recently, Carleo and Troyer [8] showed that a machine learning tool, which they called *neural-network quantum states* (NQS), can effectively and efficiently simulate quantum many-body systems in different quantum phases and for different parameters of the system. Their approach can be seen as an unsupervised neural network implementation of a variational quantum Monte Carlo method. The authors used a restricted Boltzmann machine (RBM) to calculate the ground state energy and the time evolution of quantum many-body systems such as the Ising and Heisenberg models. This work triggered a wave of interest in the design of neural network approaches to the study of quantum many-body systems [26]. This NQS method has been further explored by studying its quantum entanglement properties [13], its connection with other methods [9, 15] and its representation power [19, 23]. It has also been used to find the excited states [10], to study different models [11, 12, 25] and to aid the simulation of quantum computing [20]. Finally, as shown in [14], more powerful descriptions of quantum states can be achieved by simply increasing the depth of the neural network.

We present here an approach to finding the quantum critical points of the quantum Ising model using innate RBMs, transfer learning and unsupervised learning for NQS. We show that our approach can significantly improve the efficiency and effectiveness of a simple network like RBM.

We first propose to analytically construct restricted Boltzmann machine neural-network quantum states (RBM-NQS) for quantum states deep in each phase of the system. We refer to such RBM-NQSs as *innate* as they have innate knowledge, i.e. built-in knowledge rather than knowledge acquired by training, of the system they represent.

We then devise a transfer learning protocol across parameters of the system to improve both the efficiency and the effectiveness of the approach. We finally combine the transfer learning protocol across system parameters with a transfer learning protocol to larger sizes [41] to find the quantum critical points in the limit of infinite size.

Gale Martin, in [24], was the first to evaluate the opportunity of directly copying neural network weights trained on a particular task to another neural network with a different task to improve efficiency. This *transfer learning* approach was later improved and formalised in [2, 29, 34]. It has been applied to all kinds of learning tasks, and was shown to improve not only efficiency but also effectiveness [28, 36, 40].

We evaluate the efficiency and effectiveness of the approach for one-, two- and three-dimensional Ising models in comparison with other traditional approaches such as exact diagonalization method [37], a numerical approximation method called tensor network method [31] and a stochastic approximation method called quantum Monte Carlo method [16].

[1] School of Computing, National University of Singapore, Singapore, emails: remmy@u.nus.edu, steph@nus.edu.sg
[2] Centre for Quantum Technologies, National University of Singapore, Singapore, email: e0008987@u.nus.edu
[3] Engineering Product Development Pillar, Singapore University of Technology and Design, Singapore, emails: ryan_tan@mymail.sutd.edu.sg, dario_poletti@sutd.edu.sg
[4] Université Côte dAzur, CNRS, INPHYNI, Nice, France, emails: {frederic.hebert, mario.gattobigio}@inphyni.cnrs.fr
[5] MajuLab, CNRS-UCA-SU-NUS-NTU International Joint Research Unit, Singapore, email: cqtmc@nus.edu.sg
[6] Department of Physics, National University of Singapore, Singapore
[7] School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore
[8] Yale-NUS College, Singapore
[9] Science and Mathematics Cluster, Singapore University of Technology and Design, Singapore

The rest of the paper is structured as follows. Section 2 presents the necessary notions of quantum many-body physics, the Ising model, RBM and RBM-NQS. Section 3 presents the general approach and the algorithm for finding quantum critical points, the transfer learning protocols and the analytical construction of an initial RBM-NQS. Section 4 reports the result of the comparative performance evaluation of our approach. We conclude and highlight possible future works in Section 5.

The extended version of this paper is available in [42].

2 NEURAL-NETWORK QUANTUM STATES

2.1 Quantum many-body systems

A quantum many-body system [33] consists of a large number of interacting particles, or bodies, evolving in a discrete or continuous D-dimensional space. These particles are generally characterised by internal and external degrees of freedom. In the following, we concentrate on identical particles pinned at the nodes of a D-dimensional lattice ($D = 1, 2, 3$) and fully described by their spin internal degree of freedom which characterises their magnetic properties. The size of the system is then given by the number N of particles, the number of possible spin states per particle being n_s. In the rest of the paper, we consider one-half spins, meaning that each particle can only have $n_s = 2$ spin states.

A quantum many-body model defines how particles interact with each other or with external fields. Several prototypical models, such as the Ising model, describe the pairwise interactions of the spins of particles in addition to the interaction with external fields. The physical properties of each model depend on the respective magnitude of all these interactions, which becomes the parameters of the model.

Specifying the value of the spin for each particle gives a configuration of the system. The number of possibe configurations is, therefore, 2^N in our case.

In quantum physics, the possible physical states of a given system are described by state vectors $|\Psi\rangle$, called wave functions, living in the so-called state space. Formally, this state space is a complex separable Hilbert space and state vectors are simply linear combination of all the basis state vectors, denoted by $|x\rangle$, associated to each possible configuration x, $|\Psi\rangle = \sum_x \Psi(x)|x\rangle$. The dimension of the Hilbert space is given by the number of possible distinct configurations, which is 2^N in our case. Each complex coefficient $\Psi(x)$ is called a probability amplitude. $|\Psi(x)|^2/Z_\Psi$ gives the probability of the configuration x in the state $|\Psi\rangle$, where $Z_\Psi = \sum_x |\Psi(x)|^2$. The collection of all these probabilities defines the multinomial probability distribution of all possible configurations x of the system.

For a given grid, number of particles and external fields, the dynamics of a system is fully described by its Hamiltonian. The Hamiltonian is a Hermitian matrix of size $n_s^N \times n_s^N$ that describes how the system evolves. Furthermore, the eigenvalues of the Hamiltonian are the possible energies of the system and the corresponding eigenvectors are the only possible states in which the system can be individually found after a measurement of its energy has been performed.

The energy functional $E[\Psi]$ of a state with wave function $|\Psi\rangle$ is given in Equation (1), where E_{loc} is the local energy function of a given configuration x, as defined in Equation (2), where $H_{x,x'}$ is the entry of the Hamiltonian matrix for the configurations x and x'.

$$E[\Psi] = \sum_x \frac{|\Psi(x)|^2}{Z_\Psi} E_{loc}(x) \tag{1}$$

$$E_{loc}(x) = \sum_{x'} H_{x,x'} \frac{\Psi(x')}{\Psi(x)} \tag{2}$$

Formally, the energy functional is the expected value of the local energy. Do note that the local energy E_{loc} of any configuration x gives the average energy value of the corresponding state $|x\rangle$. Based on the variational principle in quantum mechanics, the energy functional of a given state $|\Psi\rangle$ is always larger than or equal to the lowest possible energy of the system, i.e. to the lowest eigenvalue of the Hamiltonian. It reaches this minimal value when $|\Psi\rangle$ is precisely the corresponding eigenvector called the ground state of the system.

Several methods have been developed to find the ground state. The most straightforward method is to diagonalize the Hamiltonian matrix [37, 22]. This method does not scale well as the size of the system increases. Instead, deterministic and stochastic approximation methods have been proposed and used. Tensor network methods [31] are deterministic approximation methods using variational techniques and combining the exact diagonalization with the idea of density matrix renormalisation group [38]. Quantum Monte Carlo methods [16] are stochastic approximation methods.

A phase is a region in the space of the parameters of a model in which systems have similar physical properties. In the thermodynamic limit, each possible phase is characterised by so-called order parameters that achieve different values in each phase region.

A phase transition occurs when the system crosses the boundary between two phases and the order parameters change values. When this happens, the nature and the properties of the system change qualitatively. The transition happens when the parameters of a model are varied. In the limit of infinite system size, the transition is typically described by an abrupt change in the observable physical properties or their derivatives. The term "quantum phase transition" is used for phase transitions in the ground state alone (i.e. for a system at zero temperature). The parameters of a model that correspond to this abrupt change define the quantum critical points. For finite-size systems, the transition is not abrupt but smooth. Mathematically, this means that, for a given size of the system, we need to find the inflection point of the order parameter as a function of the parameters of the system. Since it is not possible to empirically determine the parameters that yield the quantum critical point of an infinite system, it is necessary to extrapolate its limit value from a series of values simulated from systems of increasing sizes. In the remainder of the paper, when we mention a critical point, we refer to the quantum critical point.

2.2 Ising Model

We focus on the quantum transverse field Ising model, which has been studied extensively in the literature [5, 32] as it is a simple model and displays most of the qualitative features present in complex models.

The Ising model describes particles pinned on the sites of a lattice carrying a binary discrete spin. Each spin is in one of two states: up or down represented by $+1$ or -1, respectively. A configuration x is given by the value of the spin on each site: $x = (x_1, x_2, \cdots, x_N)$ where $x_i = \pm 1$.

Each particle interacts with its nearest neighbours and with an external magnetic field along the x-axis, characterised by the parameters J and h, respectively. We consider a homogeneous Ising model where the parameters are translationally invariant.

Equation (3) gives the $2^N \times 2^N$ Hamiltonian matrix of the Ising model where $neigh(\cdot)$ is a function that returns the nearest neigh-

bouring sites and the σ_i^α are the Pauli matrices where $\alpha = x, y, z$ and i indicates the position of the spin it acts upon. Only the relative strength between J and h matters. For instance, Ising model with $h = 1$ and $J = 1$ has the same static properties as a model with $h = 2$ and $J = 2$ except that the energy is doubled in the latter. Therefore, we refer to $J/|h|$ as the parameter of the system in the Ising model.

$$H = -h\sum_i^n \sigma_i^x - J\sum_i^n \sum_{j\in neigh(i)} \sigma_i^z\sigma_j^z. \tag{3}$$

We are interested in the possible magnetic phases of the system. In the paramagnetic phase, the magnetic field h dominates over the interaction J. The ground state is oriented in the x-direction and the magnetisation in the z-direction is zero. All configurations are equally probable in this state. In the ferromagnetic phase, where $J > 0$ and dominates h, the particles interact to align parallel to each other. The configurations where spins are parallel to each other (e.g. all spin-ups and all spin-downs) are the most probable ones. In the antiferromagnetic phase, where $J < 0$, neighbouring particles interact to align anti-parallel to each other. Due to the symmetry of the Ising model, the antiferromagnetic phase is equivalent to the ferromagnetic one, up to a redefinition of the directions of the spins. In particular, the transitions from paramagnetic to ferromagnetic and antiferromagnetic phases will happen at the same absolute value of $J/|h|$. Therefore, in this paper, we concentrate on paramagnetic and ferromagnetic phases and consider only positive values of h. Please refer to [42] for the details of the antiferromagnetic phases.

We look at the squared magnetisation order parameter, denoted by M_F^2 and shown in Equation (4), which shows the presence of ferromagnetism, while it is zero in the paramagnetic and antiferromagnetic phases. M_F^2 becoming non zero marks the transition point between the paramagnetic and ferromagnetic phase. We refer to this order parameter as the *ferromagnetic magnetisation* M_F^2.

$$M_F^2 = \frac{1}{N^2}\sum_{x\in\boldsymbol{x}} \frac{|\Psi(x)|^2}{Z_\Psi}\left(\sum_{i=1}^N x_i\right)^2. \tag{4}$$

An equivalent magnetisation for the antiferromagnetic phase and also magnetic correlations between different spins is detailed in [42].

The order parameter can be computed exactly by diagonalization of H to get $|\Psi(x)|^2/Z_\Psi$. However, it is intractable as the size of the system increases. We then need approximate methods such as tensor network and quantum Monte Carlo for large size systems.

For the one-dimensional Ising model, in the limit of infinite size, it is exactly known that critical points are located at $J/|h| = \pm 1$ [32]. The system is antiferromagnetic when $J/|h| < -1$, paramagnetic when $-1 < J/|h| < 1$ and ferromagnetic when $J/|h| > 1$. For the two- and three-dimensional models, the three same phases are observed in the same order, but with critical points located at $J/|h| = \pm 0.32847$ [3] and $J/|h| = \pm 0.1887$ [4], respectively, based on quantum Monte Carlo simulations.

2.3 Restricted Boltzmann machine

An RBM is an energy-based generative model [21]. It consists of a visible layer $\boldsymbol{x}$ and a hidden layer $\boldsymbol{h}$. Each one of the N visible nodes $\{x_1, \cdots, x_i, \cdots, x_N\}$ represents the value of an input. The only design choice is the choice of the number of hidden nodes. It is usual to consider a multiple, α, of the number of visible nodes. The hidden layer consists of $\alpha \times N$ hidden nodes $\{h_1, \cdots, h_j, \cdots, h_{\alpha\times N}\}$. The visible node x_i and the hidden nodes h_j are connected by the weight $W_{i,j}$. An RBM is fully described by the $N \times (\alpha \times N)$ matrix of weights.

An RBM represents the distribution p of configurations of its input layer as a function of its weights as given in Equation (5), where Z_W is the normalisation constant. A gradient descent updating the weights can train an RBM to learn the probability distribution of a set of examples that minimises the log-likelihood whether it is supervised or unsupervised. The RBM is able to sample a configuration from this multinomial distribution. When trained with a set of example configurations, the RBM learns their distribution by minimising an energy function, which is the negative log-likelihood of the distribution. This is done by Gibbs sampling with stochastic gradient descent or contrastive divergence [18].

$$p(x) = \frac{1}{Z_W}\prod_j 2\cosh\left(\sum_i x_i W_{ij}\right) \tag{5}$$

The Gibbs sampling process is as follows. From a given initial visible configuration x, for each hidden node h_j, a value is generated by sampling from the conditional probability $p(h_j \mid x)$ given in Equation (6). From this hidden configuration, for each visible node x_i, a value is generated by sampling from the $p(x_i \mid h)$ given in Equation (7).

$$p(h_j = 1|x) = \text{sigmoid}\left[2\left(\sum_i x_i W_{ij}\right)\right] \tag{6}$$

$$p(x_i = 1|h) = \text{sigmoid}\left[2\left(\sum_j W_{ij} h_j\right)\right] \tag{7}$$

2.4 Restricted Boltzmann machine neural-network quantum states

An RBM-NQS is exactly an RBM where the visible node represents one of the N particles of the quantum many-body system and its value represents the value of the spin of that particle. Each node of the RBM-NQS is a Bernoulli random variable with possible outcomes representing the two values of a spin, namely -1 or $+1$.

Instead of minimising the log likelihood of the distribution of training data, as it is generally the case for unsupervised energy-based machine learning models, RBM-NQS minimise the expected value of the local energy given in Equation (8).

$$\begin{aligned} E_{loc}(x) &= \sum_{x'} H_{x,x'}\sqrt{\frac{p(x')}{p(x)}} \\ &= \sum_{x'} H_{x,x'}\sqrt{\prod_j \frac{\cosh\left(\sum_k x'_k W_{kj}\right)}{\cosh\left(\sum_l x_l W_{lj}\right)}} \end{aligned} \tag{8}$$

In RBM-NQS, in order to minimise the energy of the system, leveraging the variational principle, the expected value of the local energy of the configurations is minimised. This makes the connection between the RBM-NQS and the Hamiltonian of the system it is trying to simulate. Indeed, Equation (8) is similar to Equation (2) where the ratio of wave functions is assumed to be the same as the square root of the ratio of their norm. Here we recall that $|\Psi(x)|^2/Z_\Psi$ is the probability of a configuration, and we stress here that the ground state of the Ising model can be chosen as a real and positive function, which allows us to write $\Psi(x) = \sqrt{p(x)/Z_\Psi}$.

The unsupervised training process does not need any example. It can rely on random configurations that it generates. The iterative minimisation process alternates the Gibbs sampling of configurations, the calculation of the expected value of their local energy and stochastic gradient descent until a predefined stopping criterion is met.

The trained RBM-NQS, as described above, is an ansatz of the function $|\Psi(x)|^2/Z_\Psi$ of the ground state for a given parameter J/h and a system of given size and dimension. We can use it to sample configurations according to $|\Psi(x)|^2/Z_\Psi$, estimate the ground state energy and estimate the order parameters.

3 FINDING THE QUANTUM CRITICAL POINTS

3.1 Overview of the approach

The approach that we consider for finding the critical points is as follows. We simulate an initial system at a selected initial parameter $J/|h|$, find its ground state and calculate the order parameter corresponding to the critical point that we are looking for. We repeat the operation increasing and decreasing the parameter with an initial step size. We are looking for an inflection point in the function of the parameter of the system that gives the value of the order parameter. We recursively reduce the step size until we identify the inflection point. This first algorithm finds the inflection point of a system of a given size.

The algorithm, therefore, receives the following input: the description of the system (its dimension and its size), the initial parameter $J/|h|$ of the system, the initial step size, the order parameter and the desired precision. The algorithm additionally stores the upper bound of the parameter of the system to look for the inflection point to make sure that the algorithm terminates if it does not find any inflection point. The algorithm terminates when the desired precision is reached or no inflection point is found.

We then repeat, as long as our computing resources reasonably allow, this algorithm for increasing sizes of the system. This is done to find the value of the critical point at the limit of infinite size of the system by the extrapolation of the inflection points.

We use RBM-NQS to simulate the system and calculate the order parameters. However, the repeated training of RBM-NQS for systems under different parameters and of increasing sizes is expensive. We devise three optimisations. The first, presented in Subsection 3.2, is the analytical construction of the innate RBM-NQS for a parameter deeply in the quantum phases to avoid being accidentally trapped in a local minimum. The second, presented in Subsection 3.3, is the use of transfer learning across parameters to avoid successive cold starts. The third, presented in Subsection 3.4, is the use of transfer learning to larger sizes again to avoid successive cold starts. Algorithm 1 shows what we used to find quantum critical points.

3.2 Construction of innate restricted Boltzmann machine neural-network quantum states

From physical understanding, we can infer the form of the probability distribution $|\Psi|^2$ of the configurations of a system if sufficiently deep in each phase, and construct an innate RBM-NQS that reproduces qualitatively the features of this distribution.

Several works have analytically or algorithmically constructed RBM-NQS, e.g. [7, 27], for effective representations of quantum many-body systems. Here we use a standard topology of RBM, and instead we analytically evaluate its weights.

Algorithm 1: Finding the Quantum Critical Point

Input: Initial and maximum parameter p and p_{max}, initial and maximum system size N and N_{max} and step size s
Output: Quantum critical point
$m_p^N \leftarrow$ construct innate RBM-NQS for the model with size N and parameter p (see Subsection 3.2).
INFLECTIONS $\leftarrow [\,]$
while $N < N_{max}$ **do**
  $p_i \leftarrow$ parameter's inflection point between $-p_{max}$ and p_{max} by transferring m_p^N to m_{p+s}^N and m_{p-s}^N (see Subsection 3.3).
  add p_i to INFLECTIONS.
  transfer $m_{p_i}^N$ to $m_{p_i}^{2N}$ (see Subsection 3.4).
  $p \leftarrow p_i, N \leftarrow 2N$.
return *extrapolation of* INFLECTIONS

If $J/|h| = 0$, there are no interactions between spins, the system is in a deep paramagnetic phase and all the configurations are equiprobable. Putting all the weights to zero gives such distribution but forbids optimisation as all gradients are identical. Therefore, we sample the weights from a normal distribution with zero mean and a small standard deviation. This construction resembles the common initialisation method of the weights of an RBM [18].

If $J/|h| \to +\infty$, the interactions between particles are dominant and the system is in a deep ferromagnetic phase. The configurations where all spins are up or all spins are down are the most probable. We then construct the weights of the RBM-NQS to ensure that the probability is maximal for these two configurations. This is achieved by setting all of the weights of each visible node to a particular hidden node to be the same and zero for the other hidden nodes. Once again, instead of using zero weights, we sample small values of the weights from a normal distribution. A similar procedure can be used for the antiferromagnetic phase when $J/|h| \to -\infty$ [42].

As mentioned earlier, in order to avoid being accidentally caught in a local minimum during the initial training of the first RBM-NQS for an arbitrary initial parameter, we choose the initial parameter to be deeply in one of the phases and construct an innate RBM-NQS. We refer to this construction as RBM-NQS-I. Additionally, we refer to the RBM-NQS starting from a cold start as RBM-NQS-CS.

3.3 Transfer learning protocol among parameters

Physically, it is expected that the wave function of systems under different but nearby values of their parameters are neighbours in the Hilbert space, although this may not be true if they are separated by a phase transition. Therefore, we expect the RBM-NQS to be similar for two systems for sufficiently nearby values of the parameters.

Following the terminology in [40], the base network is a trained or RBM-NQS-I for a value of the parameter of the system. The target network is a RBM-NQS for a different value of parameter with the same number of visible and hidden nodes. We can thus directly transfer the weights from the base network to the target network.

After transferring the weights, we trained or fine-tuned the target network until it converges to a new ground state. We expect that fewer iterations are needed for the target network to converge than it would take for a cold start initialised with a set of random weights.

We apply this parameter transfer protocol to define an algorithm to look for the inflection point of a system of a given size. We first construct an RBM-NQS-I. We then calculate the order parameter value at the ground state and we iterate with this transfer learning protocol

with adaptive step sizes until we locate the inflection point. We refer to this algorithm as RBM-NQS-IT.

3.4 Transfer learning protocol to larger sizes

Physically, it is also expected that there is a relationship between the wave function of systems with the same parameter value but of different sizes as if they were the same system at different length scales [39]. We have explored such physics-inspired transfer learning protocols in [41] and demonstrated their superiority over a cold start from both the effectiveness and efficiency points of view.

We want to find the critical points in the limit of infinite size. We expect the value of the parameter corresponding to the inflection points of a system of increasing finite sizes to converge asymptotically to this limit.

In our problem, this means that we need to transfer an RBM-NQS that has been optimised for a system with a certain size to another RBM-NQS with larger size and identical parameters.

The base network is an RBM-NQS for a given value of the parameter of the system. The target network is an RBM-NQS for the same value but for a system of larger size. The protocol needs to leverage insights in the physics of the quantum many-body system and model. The details of the protocol are given in [41].

We use this transfer learning protocol to a system of larger sizes to find the inflection point for a series of systems of increasing sizes. Instead of starting from the same initial parameter at each size of the system, we instead start from the parameter at the inflection point of the system of smaller size by using transfer learning protocol to larger sizes. We then find the inflection point at the larger size. Finally, we extrapolate the value of the critical point in the limit of infinite size. We refer to this algorithm as RBM-NQS-ITT.

4 PERFORMANCE EVALUATION

The performance evaluation is threefold. We evaluate the performance of the RBM-NQS-I construction, RBM-NQS-IT for finding the inflection point for a system of a given size and RBM-NQS-ITT for finding the critical points at the limit of infinite size in Subsection 4.1, 4.2 and 4.3, respectively. We evaluate the effectiveness, which is the accuracy of the inflection point or the critical point, and the efficiency, which is the processing time. All of the evaluations are done for systems with open boundary conditions.

The training of the RBM-NQS is done in an iterative manner. In each iteration, we take 10,000 samples to evaluate the local energy and its gradients. At the last iteration, we use these samples to calculate the order parameters. We update the weights using a stochastic gradient descent algorithm with RMSProp optimiser [17] where the initial learning rate is set to 0.001. Based on our empirical experiments, we set $\alpha = 2$ considering the efficiency and effectiveness trade-off. For RBM-NQS-CS, a random weight is sampled from a normal distribution with 0.0 mean and 0.01 standard deviation following the practical guide in [18]. For RBM-NQS-I, a random weight is sampled from a normal distribution with either 0.0 or 1.0 mean and 0.01 standard deviation as required by the construction. Note that the value of 1.0 was chosen as it results in better performance after testing a range of values between 0.1 and 1.5.

The training stops after it reaches the dynamic stopping criterion used in [41], i.e. when the ratio between the standard deviation and the average of the local energy is less than 0.005 or after 30,000 iterations. Since there is randomisation involved in the training, the value reported in the paper is an average of 20 realisations of the same calculation.

We compare this approach with the traditional methods of exact diagonalization, tensor networks and quantum Monte Carlo. For the exact diagonalization, we use the implicitly restarted Arnoldi method to find the eigenvalues and eigenvectors [22]. Our computational resources only allow us to compute exact diagonalization up to 20 particles. For the tensor network method, we use the matrix product states algorithm [31] with a bond dimension up to 1000. Both of the methods run only once since there is no randomisation involved.

The existing code of RBM-NQS is implemented in C++ with support for Message Passing Interface under a library named NetKet [6]. We ported the code into TensorFlow library [1] (available on Github [1]) for a significant speedup with the graphics processing units.

For the algorithm to find the inflection point, we choose the initial step size as 1.0 and we divide the step size by 10 at each iteration. The algorithm stops when the precision is 10^{-3}.

4.1 Construction of innate restricted Boltzmann machine neural-network quantum states

The performance evaluation of RBM-NQS-I deeply in each phase is twofold. First, we construct RBM-NQS-I without training and evaluate them. Second, we fine-tune the RBM-NQS-I until it reaches the stopping criterion and evaluate them. We evaluate the effectiveness and efficiency by comparing the value of the energy and the order parameters and by comparing the iterations needed for the training until it reached the stopping criterion with RBM-NQS-CS, respectively.

We choose $J/|h| = 0$ and $J/|h| = 3$ for the cases of deep paramagnetic and ferromagnetic phases, respectively. In the ferromagnetic case, the weights are sampled from a normal distribution with either 0.0 or 1.0 mean and 0.01 standard deviation as prescribed in Subsection 3.2.

Table 1 shows the evaluation of the RBM-NQS-CS and RBM-NQS-I for a one-dimensional system where the size of the system is 128 and the system is deep in the paramagnetic phase ($J/|h| = 0$) or the ferromagnetic phase ($J/|h| = 3$).

For $J/|h| = 0$, we observe that both the energy and the order parameter for both the RBM-NQS-CS and RBM-NQS-I without training are very close to the result of the tensor network method. When we train the RBM-NQS-I, it stops directly because it already reaches the stopping criterion. The value of the energy and order parameter are not exactly the same as the tensor network value due to the noise introduced in the weights and from the sampling process.

For $J/|h| = 3$, we observe that the results of the RBM-NQS-I are closer to the result of the tensor network method and need less iterations to converge to the stopping criterion than RBM-NQS-CS. However, the energy and the order parameters of the RBM-NQS-I without training is quite far from the result of the tensor network method. We hypothesise that this is because $J/|h| = 3$ is not deep enough in the ferromagnetic phase. Our experiments with larger $J/|h|$ with small size systems show that the value of the energy and order parameters construction is closer to the correct value [42]. Furthermore, even though the energy of the RBM-NQS-CS is quite close to the result of the tensor network method, we observe that the value of the order parameters are very far. This means that the training of RBM-NQS-CS remains stuck in a local minimum and does not converge to the ground state.

We have done similar experiments in two- and three-dimensional

[1] https://github.com/remmyzen/nqs-tensorflow

systems. We saw trends similar to the results of one-dimensional systems [42].

We showed that using RBM-NQS-I on one-dimensional systems deep in each phase is more effective and efficient than RBM-NQS-CS. Furthermore, no training is needed in the case of deep paramagnetic phase (i.e. $J/|h| = 0$). Therefore, from this point forward, we choose $J/|h| = 0$ as our initial parameter in our algorithm for finding the critical points.

Table 1. The performance evaluation of the RBM-NQS-CS and RBM-NQS-I for one-dimensional system in Ising model where the system size is 128 and parameter of the system $J/|h| = 0$ and $J/|h| = 3$. The reported value is average value over 20 realisations. The value inside the parentheses is the standard deviation.

Method	Energy	M_F^2
$J/\|h\| = 0$		
RBM-NQS-CS without training	-127.9799 (0.0029)	0.0079 (0.0001)
RBM-NQS-I without training	-127.9799 (0.0029)	0.0079 (0.0001)
RBM-NQS-CS with training	-127.9799 (0.0029)	0.0079 (0.0001)
RBM-NQS-I with training	-127.9799 (0.0029)	0.0079 (0.0001)
Tensor Network	-128.0000	0.00781
$J/\|h\| = 3$		
RBM-NQS-CS without training	-127.9061 (0.2577)	0.0078 (0.0001)
RBM-NQS-I without training	-217.5726 (0.3152)	0.2934 (0.0009))
RBM-NQS-CS with training	-372.2911 (4.7748)	0.0981 (0.1041)
RBM-NQS-I with training	-391.7046 (0.0182)	0.9658 (0.0005)
Tensor Network	-391.9119	0.9698

4.2 Finding quantum critical points for a system of a given size

We evaluate the performance of the algorithm for finding the inflection point for a system of a given size with RBM-NQS-IT.

The performance evaluation is twofold. We first provide an analysis by plotting the values of the order parameter as a function of the parameter $J/|h|$. We then evaluate the inflection point for each system's size and compare the value to other traditional methods to compare its effectiveness.

Order parameter analysis. We plot the value of the M_F^2 as a function of $J/|h|$. We use RBM-NQS-CS and RBM-NQS-IT to compute the order parameter at the ground state of each point in the space of the parameter of the system. In the limit of infinite size, there should be an abrupt change of the derivative at the critical point and the value of the order parameter should change from 0 to an increasing function. For efficiency, we compare the time needed for all of the computation.

For one-dimensional systems, we calculate the order parameters for $J/|h|$ within the range $[0, 2]$ with 0.1 intervals and for system with size $n = \{8, 16, 32, 64, 128\}$. For two-dimensional systems, we calculate the order parameters for $J/|h|$ within the range $[0, 1]$ with 0.1 intervals and for systems with sizes $n = \{2 \times 2, 4 \times 4, 8 \times 8, 16 \times 16\}$.

Figure 1 (a,b) shows the value of the ferromagnetic magnetisation M_F^2 for one-dimensional systems with RBM-NQS-CS and RBM-NQS-IT, respectively. For RBM-NQS-CS, we observe that it fails to get to the correct value before or after the inflection point for $N = 64$ and 128. This is possibly due to the network being trapped in a local minimum. For RBM-NQS-IT, we observe that the change in the derivative of the order parameter is more abrupt as we increase the size of the system as expected and the value of M_F^2 are closer to zero in one phase and closer to a function of the distance from the critical point in the other phase as we increase the size of the system.

We observe that the weights of RBM-NQS-IT do not change drastically over the parameter range explored around the transition point. This is because the magnetisation behaves smoothly across the transition point even for the large system sizes we consider here.

The results for other order parameters show similar trends as the result of the ferromagnetic magnetisation M_F^2 [42]. The result for the tensor network method, two- and three-dimensional systems are available in [42].

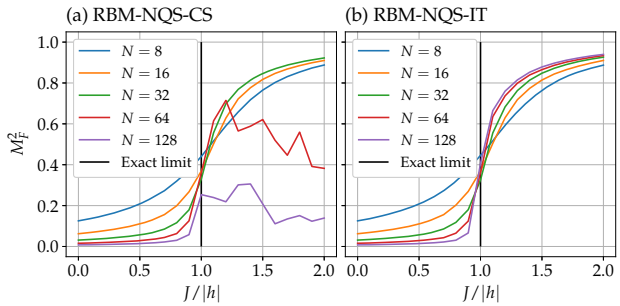

Figure 1. The ferromagnetic magnetisation M_F^2 for one-dimensional systems for $J/|h|$ within the range $[0, 2]$ with 0.1 intervals and for system with size $n = \{8, 16, 32, 64, 128\}$ with RBM-NQS-CS (a) and RBM-NQS-IT (b). The exact quantum critical point at the limit of infinite size is at $J/|h| = 1$ [32].

It takes approximately 10 minutes to compute one realisation of 128 particles with RBM-NQS-IT. Meanwhile, RBM-NQS-CS takes approximately 5 hours and our implementation of the tensor network method takes approximately 60 hours. Even though this is not a fair comparison, we show here that the RBM-NQS leveraging graphics processing units gives a significant speedup. Furthermore, RBM-NQS-IT boosts the speed even further.

Quantitative evaluation. We evaluate the inflection point for each system's size. We evaluate the performance on one-dimensional systems from $N = 8$ and doubling each time until $N = 128$. For two-dimensional systems, we start from $N = 2 \times 2$ and doubling each time until $N = 16 \times 16$.

We comparatively evaluate the effectiveness and efficiency of RBM-NQS-CS and RBM-NQS-IT. To evaluate the effectiveness, we compare the value of the inflection point at each size of the system with the tensor network method [31] and exact diagonalization for one-dimensional systems. For two-dimensional systems, we only compare with exact diagonalization.

Table 2 shows the value of the inflection point for different sizes of the system of one- and two-dimensional systems with RBM-NQS-CS, RBM-NQS-IT and tensor network method with ferromagnetic magnetisation M_F^2.

We observe that RBM-NQS-CS performs the worst overall since the value of the inflection point is far from both the tensor network and exact diagonalization methods, especially in systems of large size. It is particularly unstable in a one-dimensional system with 64 and 128 particles, as shown by a very large standard deviation. We observe that the tensor network method is closer to the exact diagonalization method for systems of small size than RBM-NQS-IT. We see that both the inflection point for RBM-NQS-IT and tensor network converge towards $J/|h| = \pm 1$, the exact critical point at the infinite size limit [32].

In two-dimensional systems, both the results of RBM-NQS-CS and RBM-NQS-IT are close to the exact diagonalization method. However, RBM-NQS-IT is closer to the exact diagonalization

method result than RBM-NQS-CS by a small margin. We believe that the performance of RBM-NQS-CS and RBM-NQS-IT is similar because of the small sizes considered, which were chosen so as to be able to compare to exact diagonalization results. The results for other order parameters and three-dimensional systems are available in [42].

It takes approximately 5 hours for one realisation to find the inflection point for innate RBM-NQS-IT for a system with a size of 128 particles. However, the absolute variance of the inflection point is relatively small, around 0.001. Therefore, in practice, one run suffices. Even though the RBM-NQS-CS takes approximately less than 1 hour, it is unstable and gives a wrong value for the inflection point. Our tensor network algorithm takes approximately 20 hours to find the inflection point.

Table 2. The value of the inflection point for one-, two- and three-dimensional systems of given sizes with RBM-NQS-CS, RBM-NQS-IT, tensor network and exact diagonalization method with ferromagnetic magnetisation M_F^2 order parameter. The value inside the parentheses is the standard deviation.

System size	RBM-NQS-CS	RBM-NQS-IT	Tensor network	Exact diag.
8	1.114 (0.009)	1.105 (0.006)	1.11	1.109
16	1.007 (0.008)	1.040 (0.005)	1.08	1.090
32	1.011 (0.009)	1.013 (0.001)	1.05	-
64	1.004 (0.009)	1 (0.001)	1.02	-
128	0.646 (0.38)	1 (0.001)	1.01	-
2×2	0.662 (0.04)	0.673 (0.05)	-	0.69
4×4	0.5 (0.0)	0.501 (0.003)	-	0.51

4.3 Finding quantum critical points at the limit of infinite size

We evaluate the effectiveness of RBM-NQS-ITT for finding the critical points at the limit of infinite size. We use the $(L, 2)-$tiling protocol defined in [41] for the transfer learning protocol to larger sizes by transferring the parameters at the inflection point of a smaller size system to a larger one.

The performance evaluation is twofold. We first provide an analysis by plotting the values of the order parameter as a function of the parameter $J/|h|$, which has been done in Subsection 4.2. We then provide an evaluation by fitting the value of the inflection point at each size of the system to show towards which value it converges in the infinite-size limit.

Order parameter analysis. We observe in Figure 1 that with RBM-NQS-IT the inflection point converges toward ± 1.0, which is the exact critical point at the limit of infinite size [32], as we increase the size of the system.

For two-dimensional and three-dimensional systems, we observe similar trends as those observed in one-dimensional systems [42]. Results for other order parameters show similar trends as those for the ferromagnetic magnetisation M_F^2 [42].

Quantitative evaluation. We evaluate the position of the critical point at the limit of infinite size by extrapolating a series of inflection points at increasing system sizes as a function of the size of the system. We fit a function of the form $f(N) = a + b\ N^c$ with nonlinear least squares, where a, b and c are the function parameters. The constraint of the parameter is $b > 0$ and $c < 0$ for ferromagnetic order parameters. The value of a approximates the value of the critical point at the limit of infinite size. We exclude the RBM-NQS-CS from this evaluation since we have shown in the previous sections that RBM-NQS-IT effectiveness is better.

Figure 2 (a) shows the evaluation of the critical point at the limit of infinite size by fitting the inflection points as a function of the size of the system in the one-dimensional model with ferromagnetic magnetisation M_F^2 order parameter. We compare the result of RBM-NQS-ITT with the tensor network method. The value of a is 0.999 and 0.923 for RBM-NQS-ITT and our tensor network algorithm, respectively.

Figure 2 (b) shows the same evaluation for systems in two dimensions. The value of a on RBM-NQS-ITT is 0.302, which is close to the value 0.32847 based on quantum Monte Carlo method [3]. The result for three-dimensional systems and with other order parameters are available on [42].

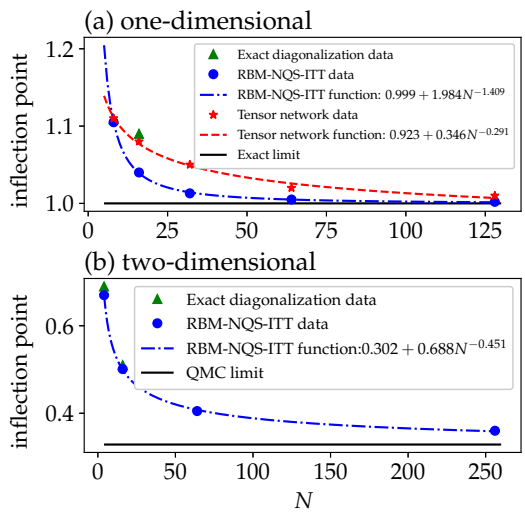

Figure 2. The evaluation of the critical point at the limit of infinite size by fitting the inflection points as a function of the size of the system in one- (a) and two- (b) dimensional models. We use the ferromagnetic magnetisation M_F^2 to find the critical point. The critical point at the limit of infinite size is at $J/|h| = 1$ [32] and $J/|h| = 0.32847$ [3] for one- and two-dimensional system, respectively.

5 CONCLUSION

We have proposed an approach to finding quantum critical points with innate restricted Boltzmann machine neural-network quantum states and transfer learning protocols. We applied the proposed approach to one-, two- and three-dimensional Ising models and in the limit of infinite size.

We have empirically and comparatively shown that our proposed approach is more effective and efficient than cold start approaches, which start from a network with randomly initialised parameters. It is also more efficient than traditional approaches.

A natural extension to this work is the study of the quantum critical exponents, which describe the behaviour of the order parameters close to the phase transitions. We also would like to further explore the opportunities to analytically and algebraically construct neural-network quantum states. Such approaches may be used to devise solutions to other problems such as characterisation of properties of

different quantum many-body systems, the study of their time evolution, as well as the study of quantum few-body systems.

ACKNOWLEDGEMENTS

We acknowledge C. Guo and Supremacy Future Technologies for support on the matrix product states simulations. This work is partially funded by the National University of Singapore, the French Ministry of European and Foreign Affairs and the French Ministry of Higher Education, Research and Innovation under the Merlion programme, project "Deep Quantum". We acknowledge support from the Singapore Ministry of Education, Singapore Academic Research Fund Tier II (project MOE2018-T2-2-142). The experiments reported in this article are performed on the infrastructure of Singapore National Supercomputing Centre (https://nscc.sg) and are funded under project "Computing the Deep Quantum".

REFERENCES

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al., 'Tensorflow: a system for large-scale machine learning.', in *OSDI*, volume 16, pp. 265–283, (2016).

[2] Jonathan Baxter, 'Theoretical models of learning to learn', in *Learning to learn*, 71–94, Springer, (1998).

[3] Henk WJ Blöte and Youjin Deng, 'Cluster monte carlo simulation of the transverse ising model', *Physical Review E*, **66**(6), 066110, (2002).

[4] Briiissuurs Braiorr-Orrs, Michael Weyrauch, and Mykhailo V. Rakov, 'Phase diagrams of one-, two-, and three-dimensional quantum spin systems', *Quantum Information & Computation*, **16**(9&10), 885–899, (2016).

[5] Sergey Bravyi and Matthew Hastings, 'On complexity of the quantum ising model', *Communications in Mathematical Physics*, **349**(1), 1–45, (2017).

[6] Giuseppe Carleo, Kenny Choo, Damian Hofmann, James E. T. Smith, Tom Westerhout, Fabien Alet, Emily J. Davis, Stavros Efthymiou, Ivan Glasser, Sheng-Hsuan Lin, Marta Mauri, Guglielmo Mazzola, Christian B. Mendl, Evert van Nieuwenburg, Ossian O'Reilly, Hugo Théveniaut, Giacomo Torlai, Filippo Vicentini, and Alexander Wietek, 'Netket: A machine learning toolkit for many-body quantum systems', *SoftwareX*, 100311, (2019).

[7] Giuseppe Carleo, Yusuke Nomura, and Masatoshi Imada, 'Constructing exact representations of quantum many-body systems with deep neural networks', *arXiv:1802.09558*, (2018).

[8] Giuseppe Carleo and Matthias Troyer, 'Solving the quantum many-body problem with artificial neural networks', *Science*, **355**(6325), 602–606, (2017).

[9] Jing Chen, Song Cheng, Haidong Xie, Lei Wang, and Tao Xiang, 'Equivalence of restricted boltzmann machines and tensor network states', *Physical Review B*, **97**(8), 085104, (2018).

[10] Kenny Choo, Giuseppe Carleo, Nicolas Regnault, and Titus Neupert, 'Symmetries and many-body excitations with neural-network quantum states', *Physical review letters*, **121**(16), 167204, (2018).

[11] Kenny Choo, Antonio Mezzacapo, and Giuseppe Carleo, 'Fermionic neural-network states for ab-initio electronic structure', *arXiv:1909.12852*, (2019).

[12] Kenny Choo, Titus Neupert, and Giuseppe Carleo, 'Two-dimensional frustrated j 1- j 2 model studied with neural network quantum states', *Physical Review B*, **100**(12), 125124, (2019).

[13] Dong-Ling Deng, Xiaopeng Li, and S Das Sarma, 'Quantum entanglement in neural network states', *Physical Review X*, **7**(2), 021021, (2017).

[14] Xun Gao and Lu-Ming Duan, 'Efficient representation of quantum many-body states with deep neural networks', *Nature communications*, **8**(1), 662, (2017).

[15] Ivan Glasser, Nicola Pancotti, Moritz August, Ivan D Rodriguez, and J Ignacio Cirac, 'Neural-network quantum states, string-bond states, and chiral topological states', *Physical Review X*, **8**(1), 011006, (2018).

[16] James Gubernatis, Naoki Kawashima, and Philipp Werner, *Quantum Monte Carlo Methods: Algorithms for Lattice Models*, Cambridge University Press, 2016.

[17] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky, 'Neural networks for machine learning lecture 6a overview of mini–batch gradient descent', (2012).

[18] Geoffrey E Hinton, 'A practical guide to training restricted boltzmann machines', in *Neural networks: Tricks of the trade*, 599–619, Springer, (2012).

[19] Zhih-Ahn Jia, Yuan-Hang Zhang, Yu-Chun Wu, Liang Kong, Guang-Can Guo, and Guo-Ping Guo, 'Efficient machine-learning representations of a surface code with boundaries, defects, domain walls, and twists', *Physical Review A*, **99**(1), 012307, (2019).

[20] Bjarni Jónsson, Bela Bauer, and Giuseppe Carleo, 'Neural-network states for the classical simulation of quantum computing', *arXiv:1808.05232*, (2018).

[21] Yann Lecun, Sumit Chopra, Raia Hadsell, Marc Aurelio Ranzato, and Fu Jie Huang, 'A tutorial on energy-based learning', in *Predicting structured data*, MIT Press, (2006).

[22] Richard B Lehoucq, Danny C Sorensen, and Chao Yang, *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*, volume 6, Siam, 1998.

[23] Sirui Lu, Xun Gao, and L-M Duan, 'Efficient representation of topologically ordered states with restricted boltzmann machines', *Physical Review B*, **99**(15), 155136, (2019).

[24] Gale Martin, *The effects of old learning on new in hopfield and backpropagation nets*, Microelectronics and Computer Technology Corporation, 1988.

[25] Kristopher McBrian, Giuseppe Carleo, and Ehsan Khatami, 'Ground state phase diagram of the one-dimensional bose-hubbard model from restricted boltzmann machines', *arXiv:1903.03076*, (2019).

[26] Roger G Melko, Giuseppe Carleo, Juan Carrasquilla, and J Ignacio Cirac, 'Restricted boltzmann machines in quantum physics', *Nature Physics*, **15**(9), 887–892, (2019).

[27] Yusuke Nomura, Andrew S Darmawan, Youhei Yamaji, and Masatoshi Imada, 'Restricted boltzmann machine learning for solving strongly correlated quantum systems', *Physical Review B*, **96**(20), 205152, (2017).

[28] Sinno Jialin Pan, Qiang Yang, et al., 'A survey on transfer learning', *IEEE Transactions on knowledge and data engineering*, **22**(10), 1345–1359, (2010).

[29] Lorien Y Pratt, 'Discriminability-based transfer between neural networks', in *Advances in neural information processing systems*, pp. 204–211, (1993).

[30] Subir Sachdev, 'Quantum phase transitions', *Handbook of Magnetism and Advanced Magnetic Materials*, (2007).

[31] Ulrich Schollwöck, 'The density-matrix renormalization group in the age of matrix product states', *Ann. Phys.*, **326**(1), 96 – 192, (2011).

[32] Sei Suzuki, Jun-ichi Inoue, and Bikas K Chakrabarti, *Quantum Ising phases and transitions in transverse Ising models*, volume 862, Springer, 2012.

[33] David J Thouless, *The quantum mechanics of many-body systems*, Courier Corporation, 2014.

[34] Sebastian Thrun and Lorien Pratt, 'Learning to learn: Introduction and overview', in *Learning to learn*, 3–17, Springer, (1998).

[35] Matthias Vojta, 'Quantum phase transitions', *Reports on Progress in Physics*, **66**(12), 2069, (2003).

[36] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang, 'A survey of transfer learning', *Journal of Big Data*, **3**(1), 9, (2016).

[37] Alexander Weiße and Holger Fehske, 'Exact diagonalization techniques', in *Computational many-particle physics*, 529–544, Springer, (2008).

[38] Steven R White, 'Density matrix formulation for quantum renormalization groups', *Phys. Rev. Lett.*, **69**(19), 2863, (1992).

[39] Kenneth G Wilson, 'Problems in physics with many scales of length', *Scientific American*, **241**(2), 158–179, (1979).

[40] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson, 'How transferable are features in deep neural networks?', in *Advances in neural information processing systems*, pp. 3320–3328, (2014).

[41] Remmy Zen, Long My, Ryan Tan, Frederic Hebert, Mario Gattobigio, Christian Miniatura, Dario Poletti, and Stephane Bressan, 'Transfer learning for scalability of neural-network quantum states', *arXiv:1908.09883*, (2019).

[42] Remmy Zen, Long My, Ryan Tan, Frederic Hebert, Mario Gattobigio, Christian Miniatura, Dario Poletti, and Stephane Bressan, 'Finding quantum critical points with neural-network quantum states', *arXiv:2002.02618*, (2020).