GHENT UNIVERSITY

Strongly Correlated Quantum Systems

(C004071A)

Fermionic Artificial Neural Networks

Authors:
Nico Fransaert
Siebe Frederix
Yannick Stulens

Promotor:
Prof. Dr. Jutho Haegeman
Supervisor:
Tom Vieijra

May 2021

Contents

1 Introduction

Many innovative technologies rely on a comprehensive understanding of the physical and chemical properties of materials. Ideally, the material behavior is deduced from first principles. Ab-initio quantum chemistry starts from fundamental quantum mechanics in order to achieve this. However, the unfavorable scaling of quantum mechanical calculations leads to a trade-off between computation time and prediction accuracy. For example, the Hartree-Fock (HF) method approximates the quantum wave function by a single Slater determinant. This method is highly efficient but typically does not reach chemical accuracy. More involved techniques that systematically increase the accuracy exist, customarily these include excitations of the HF ground state Slater. These techniques are however only suitable for relatively small system sizes — due to the unfavorable scaling, the calculations quickly become impracticable.

First principle calculations such as Hartree-Fock provide a single particle orbital basis in which the fermionic Hamiltonian in the second quantization formalism can be defined. This Hamiltonian is the central object in electronic structure calculations. The search for the ground state is then formulated as an optimization of the wave function, whereby the total energy is minimized. The fermionic Hamiltonian can be translated into a lattice problem of interacting spins by using the Jordan-Wigner mapping. Quantum spin lattice systems are routinely solved by the introduction of a variational *ansatz*, notably Jastrow-Slater wave functions and matrix product states. Based on the variational principle, the parameters of the *ansatz* are optimized to best represent the ground state wave function.

The variational form dictates the expressivity of the representation. As a consequence, an *ansatz* may fail to capture the proper entanglement properties, scale unfavorably with system dimension, or lack efficient generalization methods. Artificial neural networks (ANNs) have recently been proposed to help overcome some of these limitations [1]. Notably, the energy-based Restricted Boltzmann Machine (RBM) has been popular in this context. An approximation to the quantum wave function is obtained by optimizing the variational parameters. To extend the use of ANNs to fermionic problems, the procedure of mapping the fermionic Hamiltonian to a spin lattice and subsequently solving for the ground state has recently been applied [2]. The idea is to use the ANN as a variational representation which directly encodes the electronic configurations. From this study, it became apparent that efficient sampling of spin configurations is crucial to the performance of ANNs in the context of fermionic Hamiltonians. We propose the use of Recurrent Neural Networks (RNNs), which are able to efficiently generate independent samples by construction. Their expressive power and intrinsic sampling capabilities make RNNs excellent candidates as representations of fermionic many-body quantum systems.

2 Methods

2.1 The Fermionic Hamiltonian

In quantum chemistry, the goal is to solve the time-independent Schrödinger equation in the Born-Oppenheimer approximation [3]

$$\hat{H} |\Psi\rangle = E |\Psi\rangle , \tag{1}$$

with $|\Psi\rangle$ a n-electron wave function. We can write this n-electron wave function as a linear combination of n-electron basis functions

$$|\Psi\rangle = \sum_I c_I |\Phi_I\rangle . \tag{2}$$

In general, this n-electron basis can be built by including all possible Slater determinants formed by a complete basis of one-electron functions $\{\phi\}$ (note that this set is typically infinite). One such n-electron basis function $|\Phi_I\rangle$ is then represented by a Slater in which n one-electron orbitals $\phi_i, \phi_j, ..., \phi_k \in \{\phi\}$ are occupied [4]

$$\Phi_I(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n) = \frac{1}{\sqrt{n!}} \begin{vmatrix} \phi_i(\mathbf{x}_1) & \phi_j(\mathbf{x}_1) & \cdots & \phi_k(\mathbf{x}_1) \\ \phi_i(\mathbf{x}_2) & \phi_j(\mathbf{x}_2) & \cdots & \phi_k(\mathbf{x}_2) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_i(\mathbf{x}_n) & \phi_j(\mathbf{x}_n) & \cdots & \phi_k(\mathbf{x}_n) \end{vmatrix} . \tag{3}$$

The most basic approach to solving this problem is to use a single Slater *ansatz*, which is generally known as the Hartree-Fock (HF) method [5]. The problem is greatly reduced to the point where we have to solve a set of time-independent one-electron Schrödinger equations. The solutions to these equations are a set of orthogonal one-electron molecular orbitals. These orbitals can in turn be used as the set of one-electron basis functions $\{\phi\}$. The complete basis $\{\phi\}$ is typically infinite, which means we would have to solve an infinite set of one-electron Schrödinger equations. Because of this, the molecular orbitals are usually expanded in a finite set of atomic basis functions $\{b\}$

$$\phi_i(\mathbf{x}) = \sum_{\mu}^{N} C_{\mu}^{i} b_{\mu}(\mathbf{x}), \tag{4}$$

where N is the atomic basis set size. A well known example of such an atomic basis set is STO-3G, which we will be using throughout this work. The use of such basis sets reduces the number of equations to be solved to N (the basis set size), which in turn results in a set of N molecular orbitals $\{\phi\}_{HF}$. We can then form the HF ground state $|\Psi_{HF}^0\rangle$ by a Slater of the n molecular orbitals with the lowest energy.

More general approaches also exist. For these, the idea is to use a different *anzats* in which more than one Slater is used to approximate $|\Psi\rangle$. There are two main methods which use this approach: the Configuration Interaction (CI) method and the Coupled-Cluster (CC) method [4, 6]. Both of these methods first perform a HF calculation to build a (truncated) one-electron basis $\{\phi\}_{HF}$. In CI, the more general expansion of $|\Psi\rangle$ is made using this basis

$$|\Psi\rangle = c_0 |\Psi^0_{HF}\rangle + \sum_{i,a} c_i^a |\Psi^a_{HF,i}\rangle + \sum_{i<j,a<b} c_{ij}^{ab} |\Psi^{ab}_{HF,ij}\rangle + \cdots \,, \tag{5}$$

where $|\Psi^a_{HF,i}\rangle$ means the Slater determinant formed by replacing an occupied orbital $\phi_i, i \leq n$ in $|\Psi^0_{HF}\rangle$ with a virtual orbital $\phi_a, a \geq n$. Using this full expansion is known as the Full Configuration Interaction (FCI) method. Because of the unfavorable scaling with respect to the number of electrons, the expansion is usually truncated at single and double excitations, which is known as the CISD method. CC methods have a slightly different expansion, but the general idea stays the same. Similar truncations lead to familiar CCS (single) and CCSD (single and double) CC methods, which systematically introduce corrections to the single Slater *ansatz*.

After constructing a (truncated) basis of one-electron functions $\{\phi\}_{HF}$, we can write the many-body molecular fermionic Hamiltonian in second quantization formalism as

$$H = \sum_{i,j} t_{ij} c_i^\dagger c_j + \sum_{i,j,k,m} u_{ijkm} c_i^\dagger c_k^\dagger c_m c_j. \tag{6}$$

Here, we have defined fermionic annihilation (c_i) and creation ($c_i^\dagger$) operators with the anticommutation relation $\{c_i^\dagger, c_j\} = \delta_{i,j}$ on N fermionic modes. Furthermore, t_{ij} and u_{ijkm} are the one- and two-body integrals which are defined as

$$t_{ij} = \int \phi_i^*(\mathbf{x}) \hat{h}(\mathbf{x}) \phi_j(\mathbf{x}) d\mathbf{x}, \tag{7}$$

$$u_{ijkm} = \frac{1}{2} \int \phi_i^*(\mathbf{x_1}) \phi_k^*(\mathbf{x_2}) \frac{1}{|\hat{\mathbf{r}}_{12}|} \phi_j(\mathbf{x_1}) \phi_m(\mathbf{x_2}) d\mathbf{x_1} d\mathbf{x_2}, \tag{8}$$

with t_{ij} describing the kinetic energy and the coulomb interaction with the nuclei (expressed by the one-body operator $\hat{h}(\mathbf{x})$) and u_{ijkm} describing the coulomb repulsion between the electrons. This is the Hamiltonian we will be treating in this work.

2.2 The Jordan-Wigner Mapping

To solve the fermionic Hamiltonian using artificial neural networks as suggested in [2], a mapping to spin degrees of freedom living on a lattice is required. Several mappings have been proposed, mostly varying in simplicity and locality properties. The main idea is to preserve the canonical commutation relations. The mapping used in this paper is the Jordan-Wigner mapping defined as [7]

$$c_j \rightarrow \left(\prod_{i=0}^{j-1} \sigma_i^z \right) \sigma_j^-, \tag{9}$$

$$c_j^\dagger \rightarrow \left(\prod_{i=0}^{j-1} \sigma_i^z \right) \sigma_j^+. \tag{10}$$

Here, c_j and $c_j^\dagger$ are again the fermionic creation an annihilation operators, $\sigma_j^{(x,y,z)}$ are the Pauli operators acting on spin site j and $\sigma_j^{\pm} = \left(\sigma_j^x \pm i\sigma_j^y\right)/2$. The Jordan-Wigner mapping allows us to write the fermionic Hamiltonian as a spin Hamiltonian with the general form

$$H_q = \sum_{j=1}^{r} h_j \boldsymbol{\sigma}_j, \tag{11}$$

with h_j real and $\boldsymbol{\sigma}_j$ an N-fold tensor product of single-qubit Pauli operators $I, \sigma^x, \sigma^y, \sigma^z$. Using this transformation, we obtain a number of lattice sites equal to the number of one-electron basis functions $\{\phi\}_{HF}$.

Systems we are interested in have a fixed number of electrons, but the fermionic Hamiltonian is independent of the particle number. Therefore, sampling in the subset of the number of electrons is required. The mapping directly implies

$$n(e^-) \equiv \sum_i c_i^\dagger c_i = \sum_i \sigma_i^+ \sigma_i^-, \tag{12}$$

which commutes with the Hamiltonian.

2.3 Neural Network Quantum States

The variational principle enables the use of a variational *ansatz* that is optimized to represent the ground state of a given Hamiltonian. In such an *ansatz*, the model consists of (possibly many) parameters $\mathcal{W}$ which are systematically optimized to lower the energy $E(\mathcal{W})$. This leads to an objective function of the form

$$\min_{\mathcal{W}} \left[E(\mathcal{W})\right] \tag{13}$$

Recently, artificial neural networks (ANNs) have shown to successfully fulfill the role as variational ansatz [1]. Depending on the precise architecture of these networks, quantum states with specific properties can be properly represented [8, 9].

For a spin lattice with N sites of fermionic degrees of freedom expressed in the standard product spin basis, the Hilbert space is spanned by the set $\{|\boldsymbol{\sigma}\rangle\}$ consisting of all 2^N possible configurations. A neural network *ansatz* with input dimension N often takes one such configuration and associates this to the complex amplitude $\psi(\boldsymbol{\sigma}) \equiv \langle \boldsymbol{\sigma} | \Psi_{\mathcal{W}} \rangle$. This association can be energy-based as with the Restricted Boltzmann Machine (RBM), or can be a more direct result of the model as is the case for the Recurrent Neural Network (RNN). When searching for the ground state, the parameters $\mathcal{W}$ of the models are optimized to minimize the energy as in Eq. (13). The goal is that after the optimization, the models accurately predict the complex amplitudes $\psi(\boldsymbol{\sigma})$ of the basis states $|\boldsymbol{\sigma}\rangle$ as these are present in the actual ground state of the quantum wave function

$$|\Psi_{\mathcal{W}}\rangle = \sum_{\boldsymbol{\sigma}} \psi(\boldsymbol{\sigma}) \, |\boldsymbol{\sigma}\rangle \, . \tag{14}$$

2.3.1 Variational Monte Carlo

For neural network quantum states, the optimization is performed using a procedure called variational Monte Carlo, in which the parameters $\mathcal{W}$ are updated iteratively. At each iteration, they are adjusted as to lower the energy. This is done by calculating the gradient of the energy with respect to the parameters.

More formally, the energy expectation value

$$E(\mathcal{W}) = \frac{\langle \Psi_{\mathcal{W}} | \hat{H} | \Psi_{\mathcal{W}} \rangle}{\langle \Psi_{\mathcal{W}} | \Psi_{\mathcal{W}} \rangle} \tag{15}$$

is minimized by optimizing the network parameters $\mathcal{W}$. In general, we do not have full access to the variational wavefunction $|\Psi_{\mathcal{W}}\rangle$, and the calculation of this energy is non-trivial. To overcome this, we write the expectation value as

$$E(\mathcal{W}) = \frac{\sum_{\boldsymbol{\sigma}} \langle \Psi_{\mathcal{W}} | \boldsymbol{\sigma} \rangle \langle \boldsymbol{\sigma} | \hat{H} | \Psi_{\mathcal{W}} \rangle}{\sum_{\boldsymbol{\sigma}} \langle \Psi_{\mathcal{W}} | \boldsymbol{\sigma} \rangle \langle \boldsymbol{\sigma} | \Psi_{\mathcal{W}} \rangle}, \tag{16}$$

where we inserted resolutions of the identity in terms of the basis elements $|\boldsymbol{\sigma}\rangle$. After introducing the local energy $e_L(\boldsymbol{\sigma})$ and the distribution $p(\boldsymbol{\sigma})$ as

$$e_L(\boldsymbol{\sigma}) = \frac{\langle \boldsymbol{\sigma} | \hat{H} | \Psi_{\mathcal{W}} \rangle}{\langle \boldsymbol{\sigma} | \Psi_{\mathcal{W}} \rangle} \quad \text{and} \quad p(\boldsymbol{\sigma}) = \frac{\langle \boldsymbol{\sigma} | \Psi_{\mathcal{W}} \rangle^2}{\sum_{\boldsymbol{\sigma}} \langle \boldsymbol{\sigma} | \Psi_{\mathcal{W}} \rangle^2}, \tag{17}$$

the energy can be seen as the average of the local energy $e_L(\boldsymbol{\sigma})$ over the probability distribution $p(\boldsymbol{\sigma})$, namely

$$E(\mathcal{W}) = \sum_{\boldsymbol{\sigma}} p(\boldsymbol{\sigma}) e_L(\boldsymbol{\sigma}). \tag{18}$$

In practice, N_s configurations $\{\boldsymbol{\sigma}\}$ are sampled in order to approximate the probability distribution $p(\boldsymbol{\sigma})$, allowing us to take the mean of the corresponding local energies

$$E(\mathcal{W}) \approx \frac{1}{N_s} \sum_{n}^{N_s} e_L(\boldsymbol{\sigma}_n). \tag{19}$$

The procedure allows us to determine the energy by calculating local energies, which is done efficiently by neural network quantum states when H is sparse. Because of the sparsity of the Hamiltonian, $\langle \boldsymbol{\sigma} | \hat{H} | \Psi_{\mathcal{W}} \rangle$ consists of few terms $M_{\boldsymbol{\sigma}\boldsymbol{\sigma}'} \langle \boldsymbol{\sigma}' | \Psi_{\mathcal{W}} \rangle$. Note that the overlap $\langle \boldsymbol{\sigma}' | \Psi_{\mathcal{W}} \rangle$ is given by the neural network. The above derivation can be done for arbitrary operators, so their expectation values can be calculated similarly. In particular, this allows the gradients of the energy with respect to the variational parameters to be determined, leading to methods such as stochastic reconfiguration [10].

2.3.2 Restricted Boltzmann Machines

One of the most used artificial neural networks in quantum many-body physics is the restricted Boltzmann machine (RBM) [1]. It is a generative model consisting of the visible units σ_i (input) and hidden units h_j, which in our case take on binary values. These units, along with the biases and connections between them, define the energy of a configuration,

$$E(\boldsymbol{\sigma}, \boldsymbol{h}; \mathcal{W}) = -\sum_{i=1}^{N} a_i \sigma_i - \sum_{j=1}^{M} b_j h_j - \sum_{i=1}^{N} \sum_{j=1}^{M} w_{ij} \sigma_i h_j. \tag{20}$$

The set $\mathcal{W} = \{a, b, w\}$ contains all the variational parameters and thus defines the neural network quantum state $|\Psi_{\mathcal{W}}\rangle$. The number of hidden units determines the expressivity of the network and is characterised by the number $\alpha = M/N$ describing the ratio of hidden to visible units. This energy is then used to define the wave function by

$$\psi(\boldsymbol{\sigma}) = \langle \boldsymbol{\sigma} | \Psi_{\mathcal{W}} \rangle = \sum_{h} e^{-E(\boldsymbol{\sigma}, \boldsymbol{h}; \mathcal{W})}, \tag{21}$$

reminiscent of the Boltzmann distribution after which the network is named. The output of the network is a single complex amplitude describing the wave function overlap with the input $|\boldsymbol{\sigma}\rangle$. The network is called restricted as only connections between visible and hidden units are allowed. Explicit summation over the hidden units results in a popular way of writing the wavefunction of an RBM, namely

$$\langle \boldsymbol{\sigma} | \Psi_{\mathcal{W}} \rangle = e^{\sum_i a_i \sigma_i} \times \prod_{i=1}^{M} F_i(\boldsymbol{\sigma}), \qquad \text{where} \quad F_i(\boldsymbol{\sigma}) = 2\cosh\left[b_i + \sum_{j}^{N} w_{ij} \sigma_j \right]. \tag{22}$$

The advantage of this notation is the interpretation of the number of hidden units M, which plays a role analogous to the bond dimension of matrix product states [1].

The complex amplitudes given by the RBM Eq. (21) are used to sample configurations according to Eq. (17). Explicit calculation of $p(\boldsymbol{\sigma})$ is an intensive task that suffers from exponential scaling in system size, resulting from the summation over the total Hilbert space in the denominator. Therefore, we create a Markov chain in which configurations appear according to the probability distribution $p(\boldsymbol{\sigma})$. We use the Metropolis-Hastings algorithm which generates a sequence of configurations $\boldsymbol{\sigma}_1 \to \boldsymbol{\sigma}_2 \to \cdots \to \boldsymbol{\sigma}_n$ that are sampled according to the distribution $p(\boldsymbol{\sigma})$. At each iteration, a new configuration $\boldsymbol{\sigma}'$ is proposed and accepted with rate

$$A(\boldsymbol{\sigma}' | \boldsymbol{\sigma}) = \min\left(1, \frac{p(\boldsymbol{\sigma}')}{p(\boldsymbol{\sigma})}\right). \tag{23}$$

Notice that probability distributions appear as fractions, thereby canceling the problematic denominator of $p(\boldsymbol{\sigma})$.

2.3.3 Recurrent Neural Networks

Efficient calculation of the energy expectation value Eq. (17) is desirable when using an ANN as variational *ansatz* for quantum ground states. As described earlier, generative models require Markov chains to be constructed in order to obtain estimators. However, this procedure generates correlated samples, and is inefficient when the probability distribution is sharply peaked. To introduce the recurrent neural network, note that an important class of *stoquastic* many-body Hamiltonians have ground states $|\Psi\rangle$ with real and positive valued amplitudes in the standard product spin basis [11]. This means we can write

$$|\Psi\rangle = \sum_{\boldsymbol{\sigma}} \psi(\boldsymbol{\sigma}) \, |\boldsymbol{\sigma}\rangle = \sum_{\boldsymbol{\sigma}} \sqrt{p(\boldsymbol{\sigma})} \, |\boldsymbol{\sigma}\rangle \, , \tag{24}$$

where the distribution $p(\boldsymbol{\sigma})$ can be cast into the following form using the product rule for probabilities

$$p(\boldsymbol{\sigma}) = p(\sigma_1)p(\sigma_2|\sigma_1)\dots p(\sigma_N|\sigma_{N-1},\cdots,\sigma_2,\sigma_1). \tag{25}$$

Here, $p(\sigma_i|\sigma_{i-1},\dots,\sigma_2,\sigma_1) \equiv p(\sigma_i|\sigma_{<i})$ is the conditional distribution of σ_i, given all σ_j with $j < i$.

A recurrent neural network models the probability $p(\boldsymbol{\sigma})$ by determining the conditionals $p(\sigma_i|\sigma_{<i})$ of Eq. (25). The elementary building block of an RNN is called the recurrent cell [12]. In its most basic form, the recurrent cell maps the direct sum of an input hidden vector $\boldsymbol{h}_{i-1}$ of dimension d_h (the number of memory or hidden units) and an input (visible) vector $\boldsymbol{\sigma}_{i-1}$ to an output hidden vector $\boldsymbol{h}_i$ of dimension d_h by use of a non-linear activation function f, written as

$$\boldsymbol{h}_i = f(W[\boldsymbol{h}_{i-1};\boldsymbol{\sigma}_{i-1}] + \boldsymbol{b}). \tag{26}$$

We have as parameters the weight matrix $W \in \mathbb{R}^{d_h\times(d_h+d_v)}$, the bias vector $\boldsymbol{b} \in \mathbb{R}^{d_h}$, and initial states of the recursion $\boldsymbol{h}_0$ and $\boldsymbol{\sigma}_0$ which are fixed to constant values. With this basic recurrent cell, long distance correlations are suppressed exponentially. Therefore, an extension known as the gated recurrent cell (GRU) is used [11]. The vector $\boldsymbol{\sigma}_i$ denotes the one-hot encoding of input σ_i. By sequential computation of the conditionals, the full probability $p(\boldsymbol{\sigma})$ is obtained by

$$p(\sigma_i|\sigma_{i-1},\dots,\sigma_1) = \boldsymbol{y}_i \cdot \boldsymbol{\sigma}_i, \quad \text{where} \quad \boldsymbol{y}_i \equiv S(U\boldsymbol{h}_i + \boldsymbol{c}), \tag{27}$$

with $U \in \mathbb{R}^{d_v\times d_h}$ and $\boldsymbol{c} \in \mathbb{R}^{d_v}$ are weights and biases of a Softmax layer, and S is the Softmax activation function. An illustration of this architecture is shown in Figure 1. The full probability $p(\boldsymbol{\sigma})$ is given by

$$p(\boldsymbol{\sigma}) = \prod_{i=1}^{N} \boldsymbol{y}_i \cdot \boldsymbol{\sigma}_i, \quad \text{where} \quad \|\boldsymbol{y}_i\|_1 = \|p(\boldsymbol{\sigma})\|_1 = 1. \tag{28}$$

The norms sum over positive, real values and thus represent probability distributions. For this type of RNN, the variational parameters are $\mathcal{W} = \{W, U, \boldsymbol{b}, \boldsymbol{c}\}$, which are shared among the N cells. Therefore, the RNN is compact and independent of system size N. Deep architectures are constructed by stacking RNN cells (with intermediate activation functions), the depth being denoted by the number of layers n_l.

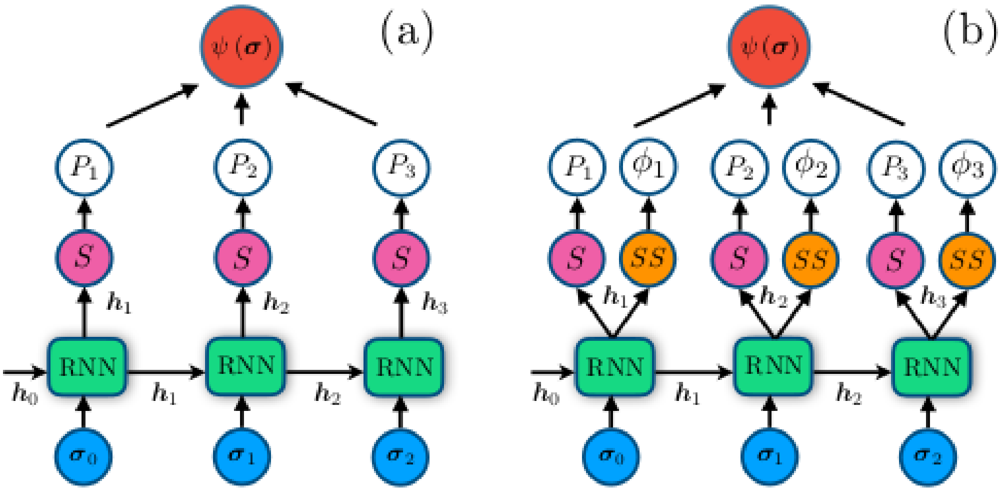

Figure 1: *Depictions of the RNN structure. a) An unfolded representation of the RNN, consisting of the input one hot vectors $\boldsymbol{\sigma}_i$, hidden vectors $\boldsymbol{h}_i$, the recurrent cell and a Softmax layer. The probability $\psi(\boldsymbol{\sigma}) = \sqrt{p(\boldsymbol{\sigma})}$ is obtained by multiplying the P_i as in Eq. 25. The probability P_i of σ_i is dependent only on σ_j with $j < i$, which is known as the autoregressive property. b) The generalization to the complex case by addition of a Softsign layer. The phase ϕ of the complex amplitude $\psi(\boldsymbol{\sigma}) = \exp(i\phi(\boldsymbol{\sigma}))\sqrt{p(\boldsymbol{\sigma})}$ is equal to the sum of the N calculated ϕ_i. Figure adapted from ref. [11].*

Sampling is performed by sequentially passing the recurrent cell, as shown in Figure 2. At each iteration, the sample σ_i is drawn from $\boldsymbol{y}_i$, which is reiterated until N configurations are obtained. It is important to note that independent successive configurations $\boldsymbol{\sigma} \to \boldsymbol{\sigma}'$ can be generated according to $p(\boldsymbol{\sigma})$. Moreover, the *normalized* probability is obtained when feeding the RNN a given configuration $\boldsymbol{\sigma}$, thereby circumventing the denominator problem in Eq. (17) and the partition function of energy-based methods.

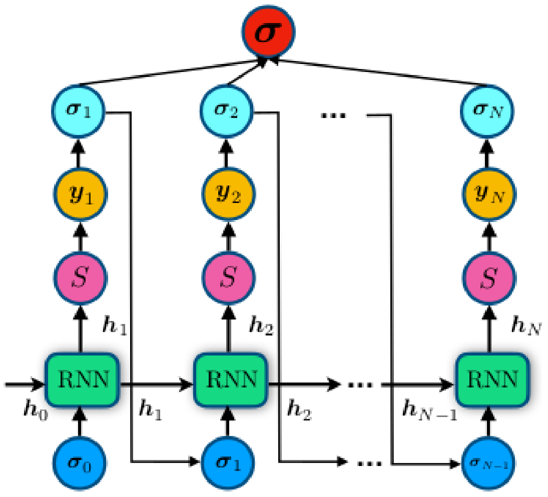

Figure 2: *An illustration of autoregressive sampling using the RNN structure. At each step i, the recurrent cell takes as input the visible one hot vector $\boldsymbol{\sigma}_i$ and hidden state $\boldsymbol{h}_i$, and computes an output hidden state $\boldsymbol{h}_{i+1}$. By passing $\boldsymbol{h}_{i+1}$ through a Softmax layer, the probability distribution $\boldsymbol{y}_{i+1}$ from which we sample σ_{i+1} is obtained [11].*

Quantum mechanical wave functions are in general a superposition of complex valued amplitudes $\psi(\boldsymbol{\sigma})$ instead of conventional probabilities $p(\boldsymbol{\sigma})$. The generalization of the RNN starts by splitting the wave function into an amplitude and phase as [13]

$$|\Psi\rangle = \sum_{\boldsymbol{\sigma}} \exp(i\phi(\boldsymbol{\sigma}))\sqrt{p(\boldsymbol{\sigma})}\,|\boldsymbol{\sigma}\rangle\,. \tag{29}$$

The phase ϕ_i for a sample σ_i is obtained by adding a Softsign layer to the model, such that

$$\phi_i = \boldsymbol{y}_i' \cdot \boldsymbol{\sigma}_i, \quad \text{where} \quad \boldsymbol{y}_i' = \pi \text{ Softsign}(U'\boldsymbol{h}_i + \boldsymbol{c}'). \tag{30}$$

For the full configuration sample $\boldsymbol{\sigma}$, we then have the phase

$$\phi(\boldsymbol{\sigma}) \equiv \sum_{i=1}^{N} \phi_i. \tag{31}$$

The amplitude part $\sqrt{p(\boldsymbol{\sigma})}$ remains computed exactly as described before. A scheme to optimize the RNN is set out in detail in ref. [11].

3 Practical Implementation

The calculations are mainly done using a combination of the following Python libraries:

- Psi4/PySCF - a collection of electronic structure programs,
- Qiskit - an open-source framework for quantum computing,
- NetKet - a framework for many-body quantum systems using ANNs,
- PyTorch - a deep learning framework.

We first define the fermionic system in PySCF and perform the HF calculation, providing us the one- and two-body integrals appearing in Eq. (6). The Qiskit library is then used to map the fermionic Hamiltonian to a spin system. The spin Hamiltonian of Eq. (11) consists of multiple terms of an N-fold tensor product. This can be represented by a string specifying the operator acting on each site along with its strength. For example, $\{Z, X, I, \dots\}$ represents σ^z acting on the first site, σ^x on the second, the identity on the third, and so on. The Jordan-Wigner mapping function of Qiskit provides these Pauli-strings and their strengths.

The majority of the calculations make use of Netket [14], a computational framework designed to study many-body quantum systems using neural networks. The RBM is defined along with an optimizer which updates the variational parameters, a sampler for the Monte Carlo Markov chain, and other training parameters. NetKet allows a Hamiltonian to be defined using the Pauli-strings obtained from Qiskit. This Hamiltonian is also used in case of the RNN, where the network is constructed using PyTorch. The code is available at https://github.com/NicoFransaert/FermionicANN.

Sampling in the n-electron subspace As the fermionic Hamiltonian commutes with the particle number operator Eq. (12), the total number of electrons n in our systems is conserved. Sampling is therefore done in the subspace of states corresponding to n electrons. For the RBM, this was realized using a sampler that proposes a new state with two spins exchanged (the Metropolis-Exchange sampler). The RNN is restricted to only generate samples from this subspace in a way closely related to how fixed magnetisation is usually imposed [11]. We note that this procedure is unbiased and conserves the normalization properties of Eq. (28).

Computational resources A grid search to discover the optimal set of hyperparameters and the calculations of the dissociation curves were performed on the Tier2 cluster of the high preformance computing facility of the VSC (www.vscentrum.be).

Reference Calculations Finally, Psi4 was used to obtain the reference energies and dissociation curves for the molecules under consideration. In the work of ref. [2], two atomic orbital bases were explored, namely STO-3G and 6-31G. Here, only the minimal basis STO-3G is explored in order to confine the scope of this work. The equilibrium molecule geometries can be obtained from the CCCBDB database (https://cccbdb.nist.gov) [15]. FCI calculations were performed to obtain the reference energies. For the sake of comparison, HF calculations were included for the dissociation curves.

4 Results & Discussion

Ground state at equilibrium geometry As a proof of principle, the ground state energy of the H_2 molecule is approximated at the equilibrium distance. The resulting energies are summarized in Table 2. Both the RBM and the RNN are optimized and their results are compared with the reference energies of HF, CCSD, and the FCI method, see Table 1. To confine the computational requirements, the calculations were performed in the minimal STO-3G basis. To find the optimal hyperparameters for our models, a grid search was performed. The hyperparameters that led to the lowest ground state energies can also be found in Table 2. Performance was estimated using the relative energy error $\Delta E_0 = |(E_0 - E_{\text{exact}})/E_{\text{exact}}|$. To minimize the error bars on the expectation value of the energy, an evaluation run is performed after the models are optimized. Note that the nuclear repulsion energy needs to be added to obtain the complete system energy.

The spin system of the H_2 molecule in the STO-3G basis consists of $N = 4$ lattice sites. The complete Hilbert space contains 16 states, but only 6 of these correspond to $n = 2$ electrons. A representative training history of the RBM is shown in Figure 3. Our energy surpasses the HF reference energy and converges towards the exact result. The relative energy error ΔE_0 shows that we obtain an accuracy higher than 1 kcal/mol, meaning chemical accuracy is achieved.

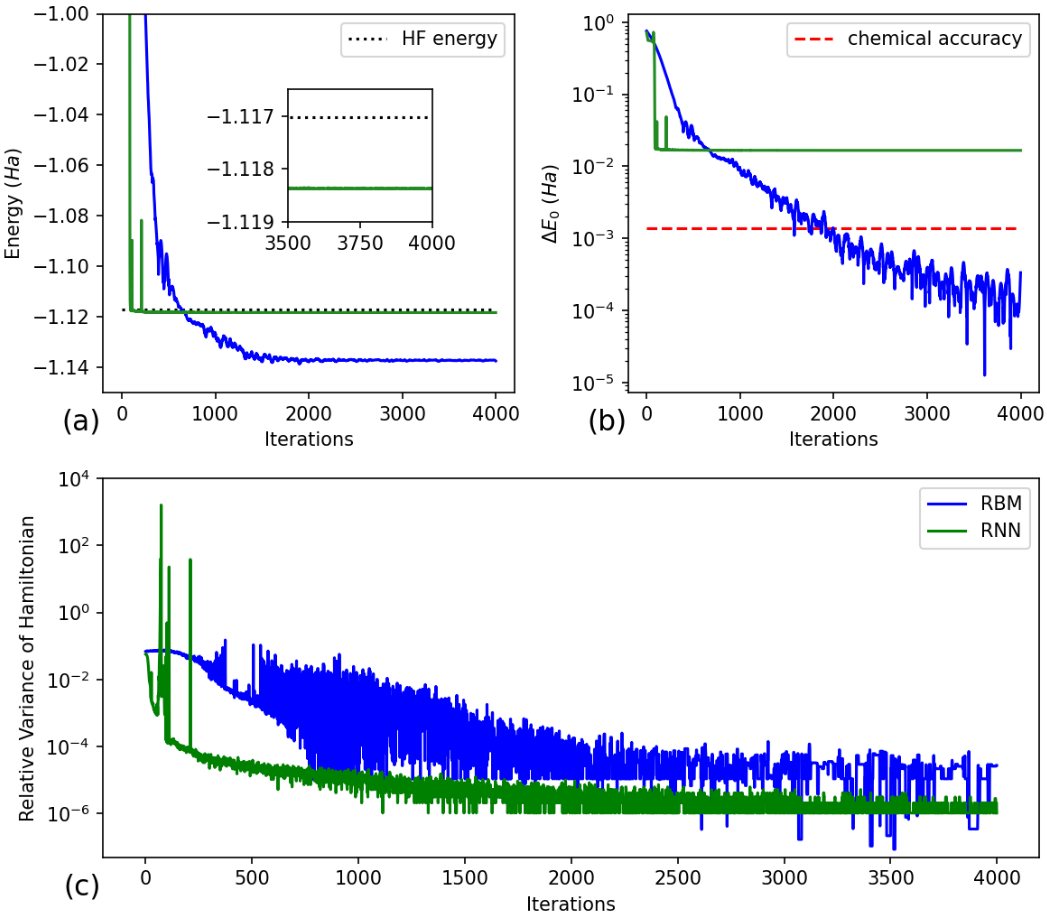

Figure 3: *Training histories of the RBM and RNN models for the H_2 molecule. a) The energy history. The RBM passes the HF energy and almost reaches the exact result, whereas the RNN converges just below the HF energy. b) The relative energy error and a threshold indicating chemical accuracy. The RBM passes this treshold after 2000 training iterations. c) The relative variance of the Hamiltonian for both models. Note that the RNN achieves lower values than the RBM, indicating that the model gets stuck in an excited state. The data is smoothed for visualization purposes.*

The training history of the RNN for the H_2 molecule is also shown on Figure 3. This model generates independent samples efficiently and without use of Markov chains, so we expect the RNN to outperform the RBM. The results however do not reflect these ideas — they show that the RNN optimization consistently leads to suboptimal ground states. We see that the RNN barely improves on the HF energy and does not pass chemical accuracy. The relative variance of the Hamiltonian $\mathrm{Var}(\hat{H}) = \frac{\langle \hat{H}^2 \rangle - \langle \hat{H} \rangle^2}{E_0^2}$ of the RNN is lower than that of the RBM, therefore we suspect that the RNN gets stuck in an excited state. It is important to note that the model generated only positive amplitudes and therefore approximated the ground state as in Eq. (24). However, the exact ground state contains complex amplitudes. Tests with the generalized RNN of Eq. (29) led to unstable optimization histories and unreliable results.

We propose several reasons to explain the deficient performance of the RNN. First, the positive valued RNN might outperform the generalized version because it is easier to find

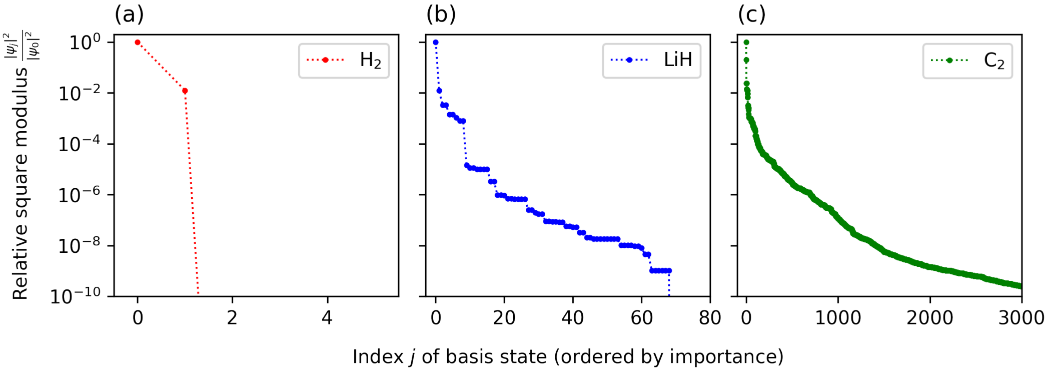

Figure 4: *The square modulus relative to the largest square modulus for the most probable configurations, ordered by decreasing probability, for a) H_2, b) LiH and c) C_2. The complex amplitudes are calculated using exact diagonalization. Note that the probability distributions are sharply peaked, leaving uniform sampling of $|\Psi_{\mathcal{W}}(\boldsymbol{\sigma})|^2$ inadequate.*

a decent local minimum in the more restricted Hilbert space. Of course, the generalized version can in principle reach lower energies than the restricted case. Second, the RNN comprises many more parameters (∼10 000) than the simple RBM *ansatz* (∼100). We expect lower energies to be reached when an *ansatz* has higher expressivity. In practice however, more parameters lead to a more complex search space, meaning the model might get stuck in bad local minima during optimization. Third, the implementation of the RNN is not perfect. Especially the energy calculations and backpropagation of the cost function slow the optimization considerably. Because of these limitations, the RNN grid search was relatively small as only a fine selection of tests could actually be run. Finally, the number of samples used to obtain estimators for the energy and gradients had to be confined to small values (∼1000 − 10 000). This constraint was mainly due to computational resources. Since the probability density is sharply peaked around the HF state (as is illustrated in Figure 4), many samples (∼100 000) are needed in order to improve significantly on the HF energy [2]. The latter is especially the case for more complex systems.

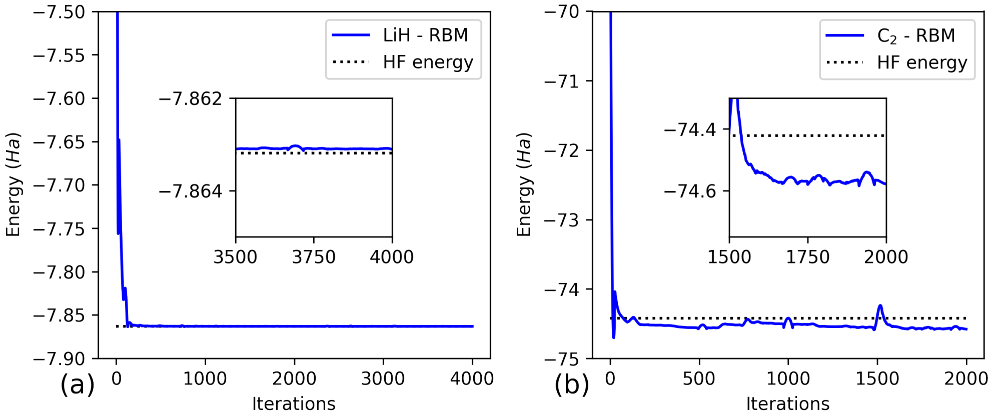

Figure 5: *Training histories of the LiH and C_2 molecule. a) For LiH, the RBM quickly reaches the HF energy, but fails to reach a lower energy due to the HF state being dominant during sampling. b) For C_2, the RBM surpasses the HF energy.*

Table 1: *Equilibrium energies (in Hartree) obtained using different methods.*

Molecule	HF	CCSD	FCI	RBM	RNN
H_2	-1.1170	-1.1373	-1.1373	-1.1373	-1.1185
LiH	-7.8632	-7.8828	-7.8828	-7.8631	
C_2	-74.4211	-74.6745	-74.6908	-74.6046	-73.2501

Analogous to the H_2 molecule explained before, more complex systems such as LiH and C_2 were investigated. This transition leads to a considerable increase in system size, even in the minimal STO-3G basis. For example, LiH is mapped to a spin system of 12 sites corresponding to a Hilbert space that has 4096 basis states. For C_2, the mapping gives a lattice of size $N = 20$ leading to a Hilbert space spanned by over a million states. Notably, the RBM performed reasonably well for the C_2 molecule whereas in case of LiH the model got stuck at the HF energy, see Figure 5. A possible explanation for this behavior is that ANNs tend to struggle with local minima when representing the wave function of small systems. This observation agrees with personal experience. The results of these optimizations and the used hyperparameters are summarized in Table 2. These results can presumably be improved upon by increasing the number of hidden units in the RBM, but this was not properly tested due to restricted computational resources. Using the same optimization procedure, the RNN was used to approximate the ground states of these more complex molecules. The resulting ground state energies could however not compete with the HF prediction.

Table 2: *Summary of the model parameters and equilibrium distances for different systems in the STO-3G basis. Distances are in Angstrom and the energies in Hartree.*

System		RBM hyperparameters				
Molecule	Eq. distance	Learning rate	α	Optimizer	N_s	Energy
H_2	0.7349	0.01	1	SGD	10000	-1.1373
LiH	1.5475	0.01	2	ADAMAX	100000	-7.8631
C_2	1.2600	0.1	1	SGD	10000	-74.6046

System		RNN hyperparameters					
Molecule	Eq. distance	Learning rate	d_h	n_l	Optimizer	N_s	Energy
H_2	0.7349	0.00025	50	1	ADAM	100000	-1.1185
C_2	1.2600	0.01	50	1	SGD	10000	-73.2501

Dissociation curve of H_2 A dissociation curve of the H_2 molecule constructed using the RBM is presented in Figure 6. The predictions of the HF and FCI methods are included for reference. We see that in the STO-3G basis, the RBM yields energies comparable to the exact result. The optimal hyperparameters were adopted from the equilibrium geometry calculations of the preceding paragraph.

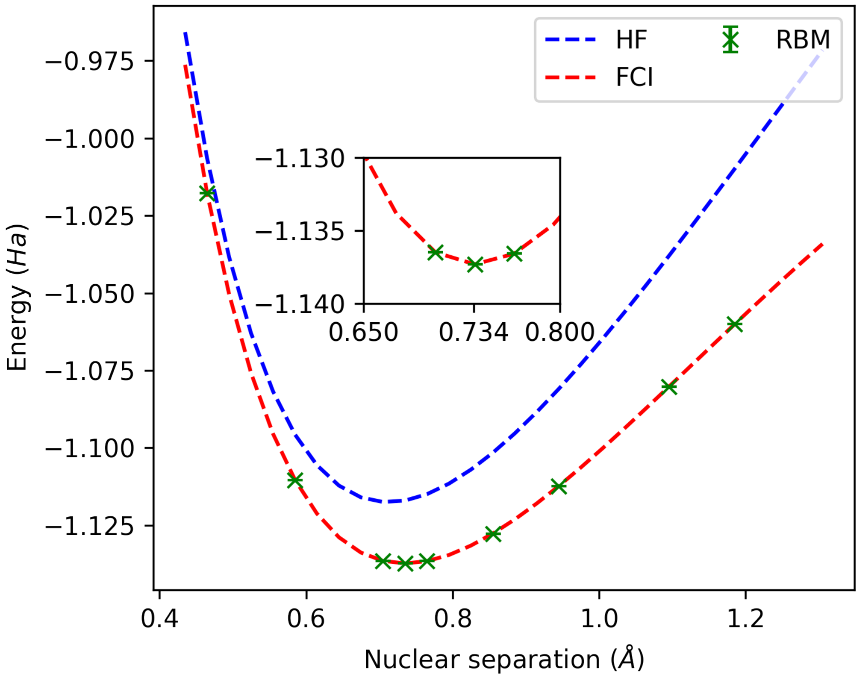

Figure 6: *The dissociation curve of the H_2 molecule in the STO-3G basis. The ground state energy of the RBM at different bond lengths (green). The HF (blue) and FCI (red) dissociation curves are shown for reference.*

This strategy can in principle be extended to more complex systems and more extensive basis sets. The main consideration is adequate sampling of the Hilbert space. The system size increases when larger basis sets are used or when simulating molecules with a larger number of electrons. With this increase in system size, the number of states in the Hilbert space increases exponentially. Since the probability density is highly peaked around the HF state, a large number of samples is needed for an adequate estimate of the energy and the gradients. Energy-based methods such as the RBM rely on Markov chains and Metropolis schemes to construct these samples according to $|\Psi_{\mathcal{W}}(\boldsymbol{\sigma})|^2$. The average acceptance probability $A(\boldsymbol{\sigma}'|\boldsymbol{\sigma})$ is typically low ($\sim$0.1%), meaning the procedure is highly inefficient. RNNs circumvent this issue, as they generate successive independent samples by construction. Since the sampling is the main bottleneck when mapping fermionic Hamiltonians to lattice systems and solving these using ANNs, we expect RNNs to be promising candidates for future research.

5 Conclusion

Summary We explored the Restricted Boltzmann Machine (RBM) and Recurrent Neural Network (RNN) as variational representations for quantum ground states. Using the Jordan-Wigner mapping, the fermionic Hamiltonian in second quantization formalism is mapped to an equivalent interacting spin system. Spin-based neural network quantum states represent a direct encoding of electronic configurations. We investigated ground state energies at the equilibrium distance for small molecules (H_2, LiH and C_2), and the dissociation curve of H_2. The Hartree-Fock (HF) and Full Configuration Interaction (FCI) methods were used as a reference. By optimizing the variational parameters of the network models, we obtained approximations to the exact result in the STO-3G basis.

The RBM showed promising results for small molecules such as H_2, reaching ground state energies close to the exact result of FCI ($\Delta E_0 \approx 10^{-4}$). Even with a small number of variational parameters ($\alpha = 1$), it managed to produce results beyond the HF method. With this proof of concept, we have shown that chemical accuracy can be exceeded by utilizing the procedure proposed by ref. [2]. The bottleneck of the procedure is adequate sampling for the estimation of expectation values and gradients. Namely, the probability distribution $\|\Psi\|$ is sharply peaked around the HF state, meaning Markov chain Monte Carlo methods are inefficient when the Hilbert space is large. More extensive basis sets and bigger molecules lead to system sizes which quickly leave this strategy impracticable.

In this work, the recurrent neural network is used to partially solve the limitations of energy-based models. RNNs autoregressively generate independent successive samples by construction, and the output probability for a given configuration is automatically normalized. As a result, no Markov chain is needed to sample according to the probability distribution $\|\Psi\|$. We expect the RNN to provide estimators more efficiently than energy-based methods such as the RBM. However, the results did not meet these expectations. The RNN achieved a ground state energy below the HF prediction for the simple H_2 molecule, but it could not compete with the RBM performance. Several explanations for this behavior have been proposed. Upon refinement of the RNN implementation, we expect the RNN to be an excellent candidate for representing fermionic many-body quantum states.

Outlook At this stage, the procedure proposed in ref. [2] is computationally feasible only for the minimal basis set STO-3G and 6-31G. The unfavorable scaling prohibits the use of extensive basis sets, meaning the calculations may lead to insufficient accuracy when complex systems are considered. However, not all the limitations are fundamental, and methods can be devised to partially circumvent them. Alternative methods to sample from the Born probability can drastically increase the performance of the procedure, thereby enabling the use of a more extensive basis set. Furthermore, only a subset of possible network architectures have been explored in this context. The introduction of the RNN is a first step in this direction, simultaneously addressing the sampling issue.

References

[1] G. Carleo and M. Troyer, “Solving the quantum many-body problem with artificial neural networks,” *Science*, vol. 355, p. 602–606, Feb 2017.

[2] K. Choo, A. Mezzacapo, and G. Carleo, “Fermionic neural-network states for ab-initio electronic structure,” *Nature Communications*, vol. 11, May 2020.

[3] M. Born and R. Oppenheimer, “Zur quantentheorie der molekeln,” *Annalen der Physik*, vol. 389, no. 20, pp. 457–484, 1927.

[4] C. David Sherrill and H. F. Schaefer, “The configuration interaction method: Advances in highly correlated approaches,” vol. 34 of *Advances in Quantum Chemistry*, pp. 143–269, Academic Press, 1999.

[5] D. R. Hartree and W. Hartree, “Self-consistent field, with exchange, for beryllium,” *Proceedings of the Royal Society of London. Series A - Mathematical and Physical Sciences*, vol. 150, no. 869, pp. 9–33, 1935.

[6] B. Jeziorski and H. J. Monkhorst, “Coupled-cluster method for multideterminantal reference states,” *Phys. Rev. A*, vol. 24, pp. 1668–1681, Oct 1981.

[7] P. Jordan and E. Wigner, “Über das Paulische Äquivalenzverbot,” *Zeitschrift fur Physik*, vol. 47, pp. 631–651, Sept. 1928.

[8] D.-L. Deng, X. Li, and S. Das Sarma, “Quantum entanglement in neural network states,” *Physical Review X*, vol. 7, May 2017.

[9] I. Glasser, N. Pancotti, M. August, I. D. Rodriguez, and J. I. Cirac, “Neural-network quantum states, string-bond states, and chiral topological states,” *Physical Review X*, vol. 8, Jan 2018.

[10] S. Sorella and F. Becca, “Sissa lecture notes on numerical methods for strongly correlated electrons,” 2015.

[11] M. Hibat-Allah, M. Ganahl, L. E. Hayward, R. G. Melko, and J. Carrasquilla, “Recurrent neural network wave functions,” *Physical Review Research*, vol. 2, Jun 2020.

[12] Z. C. Lipton, J. Berkowitz, and C. Elkan, “A critical review of recurrent neural networks for sequence learning,” 2015.

[13] G. Torlai, G. Mazzola, J. Carrasquilla, M. Troyer, R. Melko, and G. Carleo, “Neural-network quantum state tomography,” *Nature Physics*, vol. 14, pp. 447–450, May 2018.

[14] G. Carleo, K. Choo, D. Hofmann, J. E. Smith, T. Westerhout, F. Alet, E. J. Davis, S. Efthymiou, I. Glasser, S.-H. Lin, M. Mauri, G. Mazzola, C. B. Mendl, E. van Nieuwenburg, O. O’Reilly, H. Théveniaut, G. Torlai, F. Vicentini, and A. Wietek, “Netket: A machine learning toolkit for many-body quantum systems,” *SoftwareX*, vol. 10, p. 100311, 2019.

[15] R. D. Johnson, “Computational chemistry comparison and benchmark database,” Aug 2020.