

Artificial Neural Networks as Trial Wave Functions for Quantum Monte Carlo

Jan Kessler, Francesco Calcavecchia, and Thomas D. Kühne*

Inspired by the universal approximation theorem and widespread adoption of artificial neural network techniques in a diversity of fields, feed-forward neural networks are proposed as a general purpose trial wave function for quantum Monte Carlo simulations of continuous many-body systems. Whereas for simple model systems the whole many-body wave function can be represented by a neural network, the antisymmetry condition of non-trivial fermionic systems is incorporated by means of a Slater determinant. To demonstrate the accuracy of the trial wave functions, an exactly solvable model system of two trapped interacting particles, as well as the hydrogen dimer, is studied.

One class of methods for efficiently solving the many-body Schrödinger equation are variational Monte Carlo (VMC) methods,^[1,2] which are quantum Monte Carlo (QMC) schemes relying on optimizing a parametrized trial wave function according to the Rayleigh–Ritz variational principle and evaluating the inherently high-dimensional integrals by the means of importance-sampled Monte Carlo (MC) integration.^[3,4] Typically, such an approach involves the postulation of problem-specific trial wave functions, in an attempt to reduce the number of required variational parameters based on physical intuition. The apparent downside of such an ansatz are the

need of physical insights and the inflexibility of a specialized trial wave function, which is particularly problematic when, for example, phase transitions are meant to be investigated.^[5,6]


Hence, in this work, we want to consider an overall similar, but conceptionally different approach. Instead of designing a specific trial wave function, we want to propose a novel class of very general trial wave functions built from feed-forward neural networks (FFNN), that is artificial neural networks (ANN), wherein connections between the nodes do not form a cycle. The general idea is fueled by the so-called universal approximation theorem, which states that FFNNs can approximate any continuous function on a finite measure to arbitrary accuracy.^[7] Furthermore, ANNs have already proven to be a successful “black-box” approach for a variety of problems in different fields and are therefore seen as particularly well-suited for dealing with high-dimensional inputs.^[8] In turn, however, the number of variational parameters within FFNNs grows large quickly and some need for intuition is shifted toward the question of how the particle coordinates are presented to the FFNN and what activation functions are to be used in the network.

Even though, the first successful application of ANNs as wave functions for simple 1D, 2D, and 3D systems was already published more than two decades ago,^[9] the bulk of work in this direction, though only for lattice systems, is very recent.^[10–15] Particular noteworthy approaches are based on ANN representations of restricted and deep Boltzmann machines,^[16] as well as back-flow transformations.^[17,18] Very recently, however, novel QMC methods for continuous space fermionic systems based on neural networks have appeared as preprints shortly after the present work.^[19–21] Whereas the latter two are genuine multi-reference approaches based on deep neural networks using multiple determinants, we aim at locating the fermionic ground state via FFNNs with just a single Slater determinant only. Whereas the former approaches have the potential to recover a slightly higher

1. Introduction

Although the stationary Schrödinger equation describes the most basic form of quantum mechanics, finding accurate eigenstates and eigenvalues of the equation remains a formidable hard problem, at least when non-trivial realistic systems are concerned. Especially, including the important contribution of the electron correlation energy is computationally rather expensive when high accuracy is desired, which typically exhibits high-order polynomial or even exponential scaling effort with respect to the number of particles, thus rendering its application to complex systems infeasible.

J. Kessler, F. Calcavecchia, T. D. Kühne
Dynamics of Condensed Matter and Center for Sustainable Systems
Design
Chair of Theoretical Chemistry
University of Paderborn
Warburger Str. 100, D-33098 Paderborn, Germany
E-mail: tkuehne@cp2k.org
T. D. Kühne
Paderborn Center for Parallel Computing and Institute for Lightweight
Design
University of Paderborn
Warburger Str. 100, D-33098 Paderborn, Germany

 The ORCID identification number(s) for the author(s) of this article can be found under <https://doi.org/10.1002/adts.202000269>

© 2021 The Authors. Advanced Theory and Simulations published by Wiley-VCH GmbH. This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

The copyright line for this article was changed on 14 April 2021 after original online publication.

DOI: 10.1002/adts.202000269

fraction of static correlation energy, the present method shares the simplicity and numerical efficiency of single-determinant Ansätze and has fewer parameters and is hence easier to optimize. To train the employed neural networks, we devise a gradient-based VMC optimization scheme. Both components of the present method promise a favorable computational scaling with respect to the dimensionality of the problem and allows for massively parallel execution, thus providing efficient use of modern supercomputers.

The remaining of the manuscript is organized as follows: First, in Section 2, we briefly review the basic concepts of VMC and FFNNs, before introducing our novel neural network wave functions (NNWF). The application of the latter and the computational details are described in Section 3. The corresponding results are presented in Section 4, followed by our conclusions in Section 5.

2. Artificial Neural Network-Based Quantum Monte Carlo

Before presenting our novel method for approximating ground state wave functions of continuous quantum systems by employing basic FFNNs as trial wave functions, we briefly review the main concepts of VMC that is based on the Rayleigh–Ritz variational principle, thus facilitating a systematic gradient-based strategy to train the eventual NNWFs.

2.1. Variational Monte Carlo

From the non-relativistic time-independent spatial Schrödinger equation

$$\hat{H}|\Psi(\mathbf{R})\rangle = E|\Psi(\mathbf{R})\rangle \quad (1)$$

we know that the exact spectrum $\Psi_\alpha(\mathbf{R})$ of stationary wave functions is given by the eigenstates of the Hamilton operator \hat{H} , which describes the physical system of concern. The corresponding eigenvalues E_α are the energies of the respective states. The state with the lowest energy E_0 is called ground state and labeled as $\Psi_0(\mathbf{R})$.

However, for most realistic many-body Hamiltonians, it is impossible to exactly solve Equation (1) and determine the exact spectrum of eigenstates, so the problem is to find accurate approximative representations of $\Psi_\alpha(\mathbf{R})$. To approximate the ground state $\Psi_0(\mathbf{R})$ in a systematic way, we make use of the Rayleigh–Ritz variational principle

$$E_0 \leq E_T = \frac{\langle \Psi_T | \hat{H} | \Psi_T \rangle}{\langle \Psi_T | \Psi_T \rangle} = \frac{\int |\Psi_T|^2 \hat{H} \Psi_T}{\int |\Psi_T|^2} \quad (2)$$

where $\Psi_T = \Psi_T(\mathbf{R}; \omega)$ is a parametrized trial wave function with parameters ω and E_T being the corresponding energy. Since E_T is an upper bound of the true ground state energy E_0 , the variational principle immediately suggests that we can optimize Ψ_T by varying ω so as to minimize E_T . To be specific, $\Psi_T(\mathbf{R})$ is a function of $D \times N$ particle coordinates, where D is the dimensionality of the space and N the number of particles. Computing arbitrary

observables $O = \langle \Psi_T | \hat{O} | \Psi_T \rangle$ (e.g., E_T) that arise from such many-body states requires integrating over $\mathbb{R}^{D \times N}$ space, which results in solving rather high-dimensional integrals. Because of its favorable scaling with dimensionality, we employ MC integration for the numerical evaluation of these integrals.

The rightmost term of Equation (2) immediately suggests how to compute the trial wave function's energy E_T by importance sampled MC integration, in the fashion of a Metropolis–Hastings algorithm.^[22] In fact, given $\Psi_T(\mathbf{R})$, it is sufficient to compute the local energy

$$E_{loc}(\mathbf{R}_i) = \frac{\hat{H}\Psi_T(\mathbf{R}_i)}{\Psi_T(\mathbf{R}_i)} \quad (3)$$

using coordinates \mathbf{R}_i that are randomly sampled from the probability distribution $|\Psi_T(\mathbf{R})|^2$ and to average over them, that is,

$$E_T \approx E_{MC} = \frac{1}{N_{MC}} \sum_{i=1}^{N_{MC}} E_{loc}(\mathbf{R}_i) \quad (4)$$

In the equation above \approx is used to indicate that the two values are equal within the statistical uncertainty due to the stochastic sampling. Notice that because Equation (4) can be understood as a sum of independent summands, it can be evaluated in a massively parallel fashion by using multiple statistically independent random walks \mathbf{R}_i .

This scheme is typically referred to as VMC method, where we optimize the parameters of the trial wave function by means of a gradient-based stochastic optimization algorithm, described in more detail in Section 3.3.

2.2. Feed-Forward Neural Network Wave Functions

In the following, we describe how the specific NNWFs used in this work are constructed. For that purpose, we present multiple possible formulations of such wave functions, going from a straightforward and agnostic ansatz to more sophisticated and/or problem-specific variants. More precisely, we describe how we construct trial wave functions for VMC simulations from basic FFNNs. The structure of such a network is illustrated in **Figure 1**. The employed ANNs can be divided into input, hidden, and output layers, where each layer itself consists of multiple units, typically called artificial neurons or just units. Every such unit in the network signals a value to at least one other unit in the “next” layer via unidirectional connections, leading to a flow of information from the input (particle coordinates \mathbf{R}) to the output layer (wave function value $\Psi_T(\mathbf{R})$).

In the input layer, $D \times N$ input units are utilized to signal the coordinates \mathbf{R} to all units of the next layer, the first hidden layer. Additionally, there is one offset unit, signaling a static 1 in our case. Such an offset unit is present in every hidden layer as well. In the following hidden and output layers, the non-offset units are characterized by a propagation function $p(\mathbf{x}; \omega)$ and an activation function $a(p)$. For these units, the employed propagation function is a weighted sum of outputs x_i of all units from the previous layer, that is,

$$p(\mathbf{x}; \omega) = \omega \cdot \mathbf{x} = \sum_i \omega_i x_i \quad (5)$$

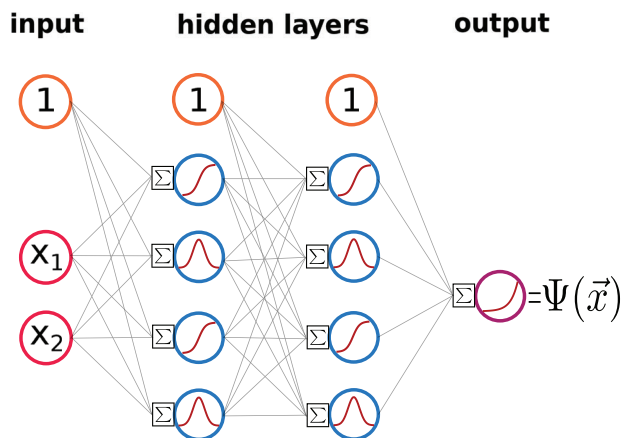


Figure 1. Illustration of a feed-forward NNWF with two input coordinates, two hidden layers, 4 + 1 hidden units each, and an exponential output activation function. The input units are depicted as red circles, offset units are orange, hidden units blue, and output units purple, respectively. The graphs inside the circles represent the activation functions. The grey lines denote the unidirectional links between units, transferring information from left to right.

where ω_i are the weights associated with each incoming connection. These weights are the variational parameters of our trial wave function. The result of the propagation function is then used as an input for the unit's activation function $a(p)$, to compute the final output value of the neuron, which is again transmitted to the next layer (if present).

While in principle most non-linear functions are usable as activation functions, after some exploration (see Appendix) we decided to use the following two hidden unit activation functions in all of our simulations:

- Tangent-sigmoid:
$$a_T(p) = \frac{2}{1 + \exp(-2p)} - 1 \quad (6)$$

- Gaussian:
$$a_G(p) = \exp(-p^2) \quad (7)$$

Notably, these two activation functions exhibit opposite symmetry and different non-linearity. Also note that they are mathematically smooth, unlike many other frequently used neural network activation functions (e.g., rectified linear functions). However, our selection is certainly just one possible choice out of many. For the output unit we employed the exponential activation function

$$a_E(p) = \exp(p) \quad (8)$$

which entails a large range of output values, while confining the output to strictly positive numbers. For detailed information about results using different other activation functions, we refer to the Appendix.

2.2.1. Non-Symmetric NNWF

For a first straightforward NNWF approach, we can simply feed the raw particle coordinates \mathbf{R} directly into the described network

via the input layer and utilize the output of the network as the wave function $\Psi(\mathbf{R})$. For simplicity, using just a single hidden layer with n_h units (including an offset), we can write such a non-symmetric wave function as

$$\Psi_{NSNN}(\mathbf{R}) = a_o \left(\omega_{o,0} + \sum_{i=1}^{n_h} \omega_{o,i} a_{h,i} \left(\omega_{h,i,0} + \sum_{j=1}^{D \times N} \omega_{h,i,j} R_j \right) \right) \quad (9)$$

where a_o is the chosen output activation function (i.e., a_L [Equation (A2)] or a_E [Equation (8)]) and $a_{h,i}$ the selected hidden layer activation functions (i.e., a_T [Equation (6)] and/or a_G [Equation (7)]). The corresponding connection weights within the output and hidden units, which will be determined by the training process, are denoted as ω_o and $\omega_{h,i}$, respectively.

A NNWF constructed in this way is (given smooth activation functions) notably a smooth function, which would be expected of a physical wave function for a finite potential energy. However, it exhibits no guaranteed symmetry under particle exchange. Yet, if the physical system of concern consists of distinguishable particles, the latter property is not directly an issue, but the implications for the application to indistinguishable particles requiring bosonic or fermionic symmetry need to be discussed. For example, a spatial wave function for (spinless) bosons is required to be strictly symmetric under arbitrary permutations of particle coordinates. However, it is known that for systems with local and exchange symmetric Hamiltonians without degeneracy and real-valued eigenstates, there is no other state with a lower energy than that of the symmetric ground state.^[23] In this case, by minimizing the energy, we would expect $\Psi_{NSNN}(\mathbf{R})$ to approximate the exchange symmetric ground state, thus becoming an approximately symmetric wave function.

Although this ansatz is rather simplistic, we still consider it as an instructive test, as it is the maximally agnostic or “naive” approach and especially because it does not entail any additional computational cost for the explicit symmetrization.

2.2.2. (Anti-)Symmetrized NNWF

Even though, it appears to be a potential option to use simple coordinate-fed FFNNs directly as wave functions - without guaranteed exchange symmetry - to approximate certain symmetric ground states, success is impossible if an approximation to an antisymmetric ground state is needed. Since the antisymmetric ground state is of higher energy than the corresponding symmetric ground state, minimizing the energy of a non-symmetric NNWF is expected to yield an approximately symmetric state. Moreover, in the special case of strictly positive valued output activation functions, the non-symmetric NNWF cannot possibly learn an antisymmetric state, by definition.

The direct way of creating guaranteed exchange symmetric or antisymmetric wave functions from asymmetric base functions would be to apply the symmetrization or antisymmetrization operators, that is,

$$\Psi_{SNN}(\mathbf{R}) = \mathcal{S} \Psi_{NSNN}(\mathbf{R}) \quad (10a)$$

$$\text{where } S = \frac{1}{N!} \sum_{P \in S_N} \hat{P} \quad (10b)$$

and

$$\Psi_{ASNN}(\mathbf{R}) = \mathcal{A} \Psi_{NSNN}(\mathbf{R}) \quad (11a)$$

$$\text{where } \mathcal{A} = \frac{1}{N!} \sum_{P \in S_N} (-1)^\pi \hat{P} \quad (11b)$$

respectively, with S_N being the symmetric group of permutations \hat{P} and π denoting the parity of the individual permutation.

For the two-particle systems concerned in this work, we can directly write the resulting NNWF as

$$\Psi_{SNN/ASNN}(\mathbf{R}) = \frac{1}{2} (\Psi_{NSNN}(R_1, R_2) \pm \Psi_{NSNN}(R_2, R_1)) \quad (12)$$

Constructing the NNWF in this way guarantees the desired exchange symmetry. However, it must be noted that the symmetrization operators implies a factorial computational scaling with the number of particles, which is prohibitive for more than a few particles. Nevertheless, for the present two-particle applications, we have included the results obtained with this ansatz, as a reference for comparison.

2.2.3. Product NNWF

So far, we have described a general method to obtain (in principle) arbitrarily exact approximations of real-valued spatial ground state wave functions from first-principles by taking only exchange symmetry into account. No further considerations about the physical system of concern were implemented into the construction of our trial wave functions. While such a general and agnostic method has its own merits, we would not want to stay away from more specialized variants, at least when they prove to be simple to realize and more efficient.

One possible idea in this direction is to utilize a $\Psi_{NSNN}(\mathbf{R})$ merely as a modulating NN-based Jastrow factor together with an imposed problem-specific determinantal wave function $\Phi(\mathbf{R})$, to form a product NNWF

$$\Psi_{PNN}(\mathbf{R}) = \Psi_{NSNN}(\mathbf{R}) \Phi(\mathbf{R}) \quad (13)$$

The determinantal part $\Phi(\mathbf{R})$ of $\Psi_{PNN}(\mathbf{R})$ could in principle be freely chosen, but we expect better results when it is already similar to the ground state that is to be approximated. Furthermore, $\Phi(\mathbf{R})$ directly prescribes the sign of the full product NNWF $\Psi_{PNN}(\mathbf{R})$, given that a strictly positive valued $\Psi_{NSNN}(\mathbf{R})$ is used. Then, $\Psi_{PNN}(\mathbf{R})$ has the same sign as $\Phi(\mathbf{R})$ for all \mathbf{R} and is equal to zero only at the same nodal surface $\{\mathbf{R} : \Phi(\mathbf{R}) = 0\}$. This property effectively restricts the set of wave functions that such a product NNWF can approximate.

To approximate bosonic ground states, we consider a symmetric Hartree-product of identical single-particle orbitals $\phi(\mathbf{r})$ as the determinantal part of the NNWF, leading to

$$\Psi_{BPNN}(\mathbf{R}) = \Psi_{NSNN}(\mathbf{R}) \phi(\mathbf{r}_1) \phi(\mathbf{r}_2) \dots \quad (14)$$

where \mathbf{r}_i depict the coordinates of all individual particles. We will refer to $\Psi_{BPNN}(\mathbf{R})$ as bosonic product NNWF.

For fermionic ground states, however, we use an antisymmetric Slater determinant Φ_{SD} of given single-particle orbitals $\phi_i(\mathbf{r})$ as the determinantal part of the eventual NNWF, that is,

$$\Psi_{FPNN}(\mathbf{R}) = \Psi_{NSNN}(\mathbf{R}) \Phi_{SD}(\mathbf{R}) \quad (15)$$

We denote $\Psi_{FPNN}(\mathbf{R})$ as fermionic product NNWF. As already alluded to above, given a strictly positive non-symmetric NNWF, the nodes of $\Psi_{FPNN}(\mathbf{R})$ are defined by $\Phi_{SD}(\mathbf{R})$. This is to say that as long as $\Phi_{SD}(\mathbf{R})$ does not change during the optimization, the nodes are fixed. If the nodal surface should be variable instead, the considered orbitals are required to contain variational parameters to be optimized together with the FFNN weights. One known possibility in this direction is the use of backflow orbitals.^[24–30] However, for the specific applications of the present work, it suffice to rely on a fixed node ansatz.

We find it important to mention that although at this time we have no mathematical proof at hand which would guarantee that the variational principle holds for $\Psi_{FPNN}(\mathbf{R})$ when using a non-symmetric NNWF, that is, there is no asymmetric state reachable with a truly lower energy than the exactly antisymmetric state. Nevertheless, we decided to test the product NNWF formulation in our applications because it promises to solve the “antisymmetry problem” at negligible computational cost compared to the previously discussed full antisymmetrization, while still retaining the simplicity of raw coordinates \mathbf{R} .

2.2.4. Symmetric-Featured NNWF

Another opportunity for improvement concerns the way information is fed to the actual FFNN. Until now, we simply used all $D \times N$ raw particle coordinates \mathbf{R} as input, but this choice might not be ideal in all cases. For example, considering the exchange symmetry, it is easy to see that the raw coordinates contain some redundant information, since for the wave function it does not matter if particle 1 is at \mathbf{r}_1 and particle 2 at \mathbf{r}_2 , or the other way around. Instead, using a symmetric representation of all particle coordinates would ensure the exchange symmetry of the NNWF and, at the same time, potentially increase its accuracy. In general, an inspection of the specific Hamiltonian might lead to more efficient representations of all particle coordinates. For more complex systems, however, multi-particle features that are invariant with respect to translational and rotational symmetries may be essential.^[31–33]

Here, we propose and implement the possibility of such representations by replacing the $D \times N$ particle coordinates \mathbf{R}_j in Eq. 9 with n_f functions $f_j(\mathbf{R})$ (“features”), resulting in

$$\Psi_{SFNN}(\mathbf{R}) = a_o \left(\omega_{o,0} + \sum_{i=1}^{n_h} \omega_{o,i} a_{h,i} \left(\omega_{h,i,0} + \sum_{j=1}^{n_f} \omega_{h,i,j} f_j(\mathbf{R}) \right) \right) \quad (16)$$

for a network with only a single hidden layer.

In this work, we want to confine ourselves to exchange symmetric sets of features, which we will introduce in Section 3, and

Table 1. An overview of NNWF types and their properties, where imposed symmetry refers to an exchange symmetry that is guaranteed by construction and expected symmetry is an (approximate) exchange symmetry that is expected to emerge from energy minimization.

NNWF name	Acronym	Imposed/Expected symmetry
Non-symmetric	NSNN	None/Bosonic
Symmetrized	SNN	Bosonic/Bosonic
Antisymmetrized	ASNN	Fermionic/Fermionic
Bosonic product	BPNN	None/Bosonic
Fermionic product	FPNN	None/Fermionic
Symmetric-featured	SFNN	Bosonic/Bosonic
Symmetric-featured bosonic product	BP-SFNN	Bosonic/Bosonic
Symmetric-featured fermionic product	FP-SFNN	Fermionic/Fermionic

therefore refer to such NNWFs as symmetric-featured NNWFs. Notably, employing a symmetric-featured NNWF as the Jastrow part of a bosonic or fermionic product NNWF yields a strictly symmetric or antisymmetric wave function. Hence, we will refer to a symmetric-featured product NNWF as bosonic/fermionic product NNWFs that are defined as

$$\Psi_{BP-SFNN}(\mathbf{R}) = \Psi_{SFNN}(\mathbf{R}) \phi(\mathbf{r}_1) \phi(\mathbf{r}_2) \dots \quad (17)$$

or

$$\Psi_{FP-SFNN}(\mathbf{R}) = \Psi_{SFNN}(\mathbf{R}) \Phi_{SD}(\mathbf{R}) \quad (18)$$

respectively.

At this point we want to note that the set of $f_j(\mathbf{R})$ must be chosen with some care. If relevant information is lost here, the symmetric-featured NNWF might not be able to approximate the true ground state anymore, no matter how large the employed FFNN is. Also important, the continuity and differentiability of our symmetric-featured NNWF will depend on the respective properties of the functions $f_j(\mathbf{R})$. In the following, we present explicit problem-specific realizations of features $f_j(\mathbf{R})$ in Section 3.

An overview of all the introduced NNWF types can be found in Table 1.

3. Applications

3.1. 2-Particle Harmonic Trap with Soft-Core Interactions

As a first application of our NNWFs, we selected a toy model system of two particles in a 1D harmonic potential that are interacting via a potential of finite-range and constant value. Such a system can be described by the Hamiltonian

$$\hat{H} = -\frac{(\partial_{x_1}^2 + \partial_{x_2}^2)}{2} + \frac{x_1^2 + x_2^2}{2} + \mathcal{V}(x_1 - x_2) \quad (19)$$

where we are using atomic units, harmonic oscillator frequency of 1 a.u. and the soft-core potential

$$\mathcal{V}(x) = \begin{cases} V, & \text{if } |x| < a \\ 0, & \text{if } |x| \geq a \end{cases} \quad (20)$$

This model is a strongly correlated system, but still was shown to be exactly solvable for both bosonic and fermionic symmetry.^[34] The outlined recipe to compute the exact solution allows us to obtain exact energy eigenvalues and eigenstates to validate our own simulation results. The soft-core interactions between the particles can be either attractive or repulsive, and both the potential range and strength can be chosen freely, thus providing a rich testing environment for our various NNWFs.

While the Hamiltonian of Equation (19) is all that is required to begin our WF optimization of the simple $\Psi_{NSNN}(\mathbf{x})$ or (anti-)symmetrized $\Psi_{S/AS}(\mathbf{x})$, for the product NNWFs $\Psi_{BP}(\mathbf{x})$ and $\Psi_{FP}(\mathbf{x})$ single-particle orbitals are required. To that extent, we consider the non-interacting system (i.e., $V = 0$), namely a harmonic oscillator, which has well-known single-particle eigenstates given by the Hermite functions ϕ_n . Relevant for us are the first two orbitals

$$\phi_0(x) = \pi^{-\frac{1}{4}} e^{-\frac{1}{2}x^2}, \quad \phi_1(x) = \sqrt{2} \pi^{-\frac{1}{4}} x e^{-\frac{1}{2}x^2} \quad (21)$$

From the first or both of these orbitals, we build the determinantal parts of our bosonic or fermionic product NNWFs, as in Equations (14) or (15), respectively.

In the case of symmetric-featured NNWFs we employ

$$f_1 = x_1^2 + x_2^2, \quad f_2 = (x_1 - x_2)^2 \quad (22)$$

as input of the FFNN, instead of the raw coordinates \mathbf{x} . For the a Hamiltonian, all relevant information are retained in this representation, while it guarantees bosonic exchange symmetry, spatial symmetry corresponding to the Hamiltonian, and is mathematically smooth. We use this representation both for our symmetric-featured simple NNWFs and for the Jastrow part of symmetric-featured bosonic/fermionic product NNWFs.

3.2. H₂ Molecule

As a more realistic application, we study the hydrogen dimer H₂ within the Born–Oppenheimer approximation, that is, we consider the electronic Hamiltonian (in atomic units)

$$\hat{H} = -\frac{1}{2}(\nabla_{\mathbf{r}_1}^2 + \nabla_{\mathbf{r}_2}^2) - \sum_{i,j=1}^2 \frac{1}{|\mathbf{r}_i - \mathbf{R}_j|} + \frac{1}{|\mathbf{r}_1 - \mathbf{r}_2|} + \frac{1}{|\mathbf{R}_1 - \mathbf{R}_2|} \quad (23)$$

where \mathbf{r} denote the variable electronic coordinates and \mathbf{R} the static protonic coordinates.

Although this system consists of electrons, which are fermions, its spatial wave function must be exchange symmetric whenever the spin wave function is antisymmetric (singlet state). In fact, the state with the lowest energy is a singlet and therefore requires an exchange symmetric spatial wave function. Hence, we may either directly employ a simple NNWF, or use our bosonic product NNWF. For the latter, we use as single-particle orbital the bonding molecular orbital

$$\phi(\mathbf{r}) = e^{-|\mathbf{r}-\mathbf{R}_1|} + e^{-|\mathbf{r}-\mathbf{R}_2|} \quad (24)$$

where \mathbf{r} are the single electron coordinates and \mathbf{R}_i the two protonic coordinates.

Unlike for the previously discussed harmonic soft-core trap, the ground state of the hydrogen molecule cannot be obtained analytically. Instead, the total energies, as computed by full configuration-interaction method, are used as a reference for comparison.^[35]

3.3. Adam-Based VMC Optimization

Since a NNWF may contain hundreds if not thousands of variational parameters, we aim to adjust them in an effort to approximate the true ground state wave function as closely as possible, via an algorithm. A notable complication are the unavoidable statistical errors present in both our cost function and their respective gradients. To deal with this high-dimensional stochastic optimization problem, we employ the popular algorithm Adam.^[36] It is a gradient-based optimization algorithm suitable for noisy gradients that internally builds up first and second order weight momenta and provides an adaptive learning rate for each of our weights ω_i . We employ the algorithm as suggested by its authors, including the usage of an exponential moving average to obtain the optimized weights (see Section 7.2 in ref. [36]), leading to a reduced influence of noise on the final result.

As cost function for the optimization, we compute the energy E_T according to Equation (4) and to prevent overfitting, we add a (small) L_2 -regularization term for the variational parameters, that is,

$$C = E_T + \frac{\lambda_r}{n_\omega} \sum_{i=1}^{n_\omega} \omega_i^2 \quad (25)$$

where n_ω is the number of parameters and λ_r a factor to control the magnitude of the term.

The gradient of the energy E_T can be derived analytically as

$$\begin{aligned} \partial_{\omega_i} E_T &= \partial_{\omega_i} \frac{\langle \psi | H | \psi \rangle}{\langle \psi | \psi \rangle} \\ &= 2 \left(\frac{\langle \psi | H | \partial_{\omega_i} \psi \rangle}{\langle \psi | \psi \rangle} - E_T \frac{\langle \psi | \partial_{\omega_i} \psi \rangle}{\langle \psi | \psi \rangle} \right) \end{aligned} \quad (26)$$

This means that in our MC integration scheme, we can compute the gradients G_i of our cost function C as

$$\begin{aligned} G_{MC,i} &= \frac{2}{N_{MC}} \left[\sum \left(E_{loc} \frac{\partial_{\omega_i} \Psi}{\Psi} \right) - E_{MC} \sum \frac{\partial_{\omega_i} \Psi}{\Psi} \right] \\ &+ 2 \frac{\lambda_r}{n_\omega} \omega_i \end{aligned} \quad (27)$$

where the summands of sums $\sum^{N_{MC}}$ are evaluated along the importance sampled random walk, as discussed in Section 2.1.

Throughout the present work, this Adam-based optimizer is used to minimize the energy of all here proposed FFNN-based trial wave functions for a given Hamiltonian.

4. Computational Details

In all of our simulations, the following basic strategy is employed to optimize the NNWF:

1. Least-squares fit of the NNWF to an initial-guess solution
2. Use of our Adam-based VMC scheme to iteratively optimize the NNWF
3. Identify the result with the lowest variational energy

First, the NNWF is initialized to a well-behaved state using a relatively fast least-squares fitting procedure before the actual VMC optimization is performed. Even though this procedure is not strictly necessary, it increases efficiency and stability of the subsequent VMC algorithm, especially when the chosen initial state is already close to the true ground state solution. Except for product-type NNWFs, a product of Gaussian single-particle orbitals is employed to perform this step. Such an initial state appears to be a straightforward first guess for systems with localized ground states. For product-type NNWFs, however, we skipped the first step and started with weights initialized to small random numbers. Note that for technical reasons we did not use any symmetrizing operators during the fitting procedure, even when they were used in the following VMC optimization.

Starting with the pre-fitted FFNN to represent the NNWF, we conducted an Adam-based optimization, which will be described in the following, with parameters $\alpha = 0.002$, $\beta_1 = 0.9$, $\beta_2 = 0.9$, $\epsilon = 10^{-8}$ (notation as in ref. [36]) and regularization factor $\lambda_r = 0.005$. The variational parameters of the NNWF were initially updated according to the Adam algorithm until the last 100 optimization steps yielded constant energy, with respect to integration errors. With the resulting averaged variational parameters, a final MC evaluation of the respective energy was carried out. Afterward, the Adam-based VMC optimization procedure was restarted, beginning with the final result from the previous optimization run.

Eventually, out of the individual resulting energies of all chained optimization runs, the one with the lowest upper energy confidence bound

$$E_{ub} = E_{MC} + 2 \sigma_{MC} \quad (28)$$

with σ_{MC} being the integration error estimation, is selected. This is done, since the last optimization run is not necessarily the one with the lowest variational energy, due to the stochastic nature of the optimization. These NNWFs were then employed within extensive MC sampling runs to evaluate the energies and all other observables reported in this paper.

For each optimization step, the energies and gradients were computed based on 4×10^6 , the final energy evaluation of each chained optimization run based on 8×10^7 and the eventual observables based on 10^9 MC iterations, which were distributed among 16 CPU cores in all cases. For the MC sampling, we used all-particle moves proposed from a uniform distribution, where the maximum step-size was calibrated to yield an acceptance probability of $\approx 50\%$ during 2500 equilibration steps per thread. For better stability when using NNWFs, we confined the walkers within a periodic interval of length $-10 < x < 10$ Bohr, which is

large enough to still allow to properly simulate non-periodic systems of interest.

All FFNNs employed here contained two hidden layers, each with 12 hidden units and 1 offset unit. A comparison of results for different network sizes can be found in Figure A3 within the Appendix. To realize our NNWF optimization in practice, we wrote our own C++ libraries for all involved tasks. All relevant libraries are available in the DCM-UPB/NNVMC repository on GitHub (<https://github.com/DCM-UPB/NNVMC>).

5. Results

5.1. 2-Particle Harmonic Trap with Soft-Core Interaction

Using the previously described optimization scheme, we applied all of our NNWF methods to the soft-core harmonic trap system, for various choices of Hamiltonian parameters a (potential range) and V (potential strength). Exemplary, **Figure 2** shows the resulting energy values for simple bosonic (panel (a)) and fermionic (panel (b)) product NNWFs, in comparison with the exact bosonic and fermionic ground state energies computed based on the exact solutions presented in ref. [34]. At least on the scale of these plots, the energies obtained from the NNWFs perfectly match the exact energies, for all considered Hamiltonian parameters. The same holds true for all other employed NNWFs.

For a more in-depth analysis of the results, we categorize all of our NNWF methods we have proposed here into two groups:

1. Non-symmetric NNWFs without strict exchange symmetry, that is, plain NNWF, as well as simple bosonic and fermionic product NNWFs.
2. (Anti-)Symmetric NNWFs with strict bosonic or fermionic exchange symmetry, that is, symmetrized and antisymmetrized NNWFs, symmetric-featured NNWF, as well as symmetric-featured bosonic and fermionic product NNWFs.

For both groups, we will always display the resulting energy residuals $E_T - E_0$, that is, the difference between our VMC energy result E_T and the exact ground state energy E_0 . However, because the first group is technically prone to exhibit significant asymmetry, for these NNWFs, we will also check positional observables and a measure of how well the desired exchange symmetry is realized.

5.1.1. Energy Residuals

Beginning with the group of asymmetric NNWFs, we display their energy residuals in **Figure 3a**. Note that the residual axis is scaled logarithmically to compensate for the large range of values. Furthermore, for any product-type NNWF, the interaction-free case $V = 0$ is not shown, because the imposed determinantal part of our product NNWFs is already exactly the ground state of that system.

In **Figure 3a**, we are mainly looking for effects of using the simple bosonic-product NNWF over the simple non-symmetric NNWF. Overall, it appears that there is no clearly superior ansatz among the two. However, the product ansatz yields consistently similar or even lower energies, whenever the soft-core potential

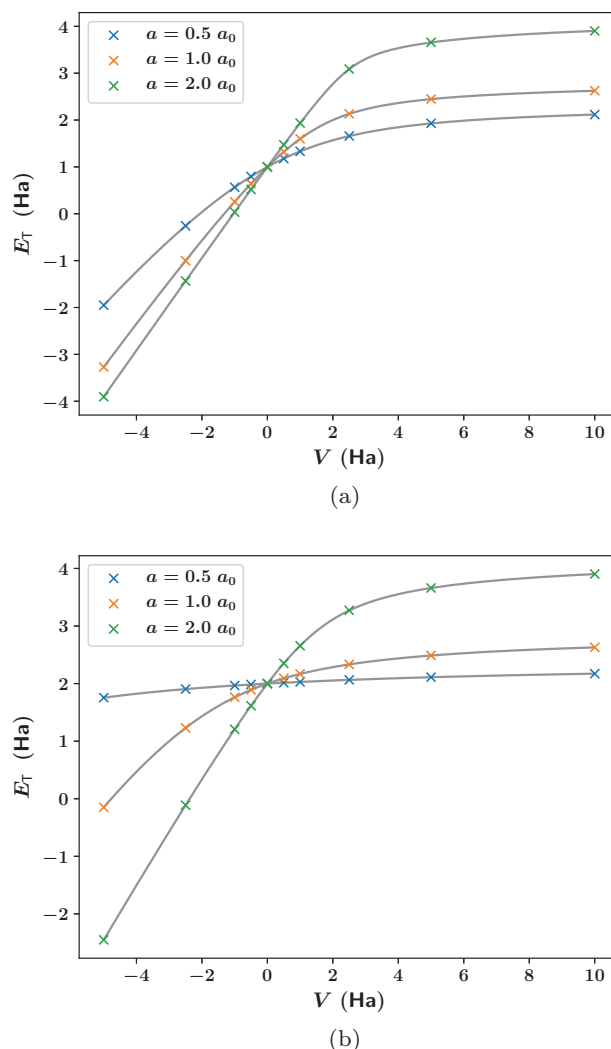


Figure 2. Energies as obtained using a) simple bosonic product and b) simple fermionic product NNWFs, respectively. For comparison, the corresponding exact bosonic and fermionic ground state energies are shown (solid grey lines), for various potential ranges a and potential strengths V .^[34]

is attractive, or low by absolute value. In turn, it yields similar or higher energies for strong repulsive interactions. This behavior can be understood as a verification of our assumption that the product-type NNWF perform better when their imposed determinantal part is more similar to the true ground state. For our bosonic product NNWF, we are using the bosonic ground state of a two-particle harmonic oscillator, which with some hand-waving can be said to be a first guess that tends to fit better when the inter-particle interaction is attractive or small and worse for strong repulsive interactions (also see **Figure 3d**).

Looking at the simple fermionic product NNWF, which uses the fermionic ground state of a two-particle harmonic oscillator, we get the opposite picture instead (as expected): It tends to perform better for repulsive interaction than for attractive interactions. Especially noteworthy is that the simple fermionic product NNWF predict accurate (residual $< 10^{-3}$ Ha) fermionic energies

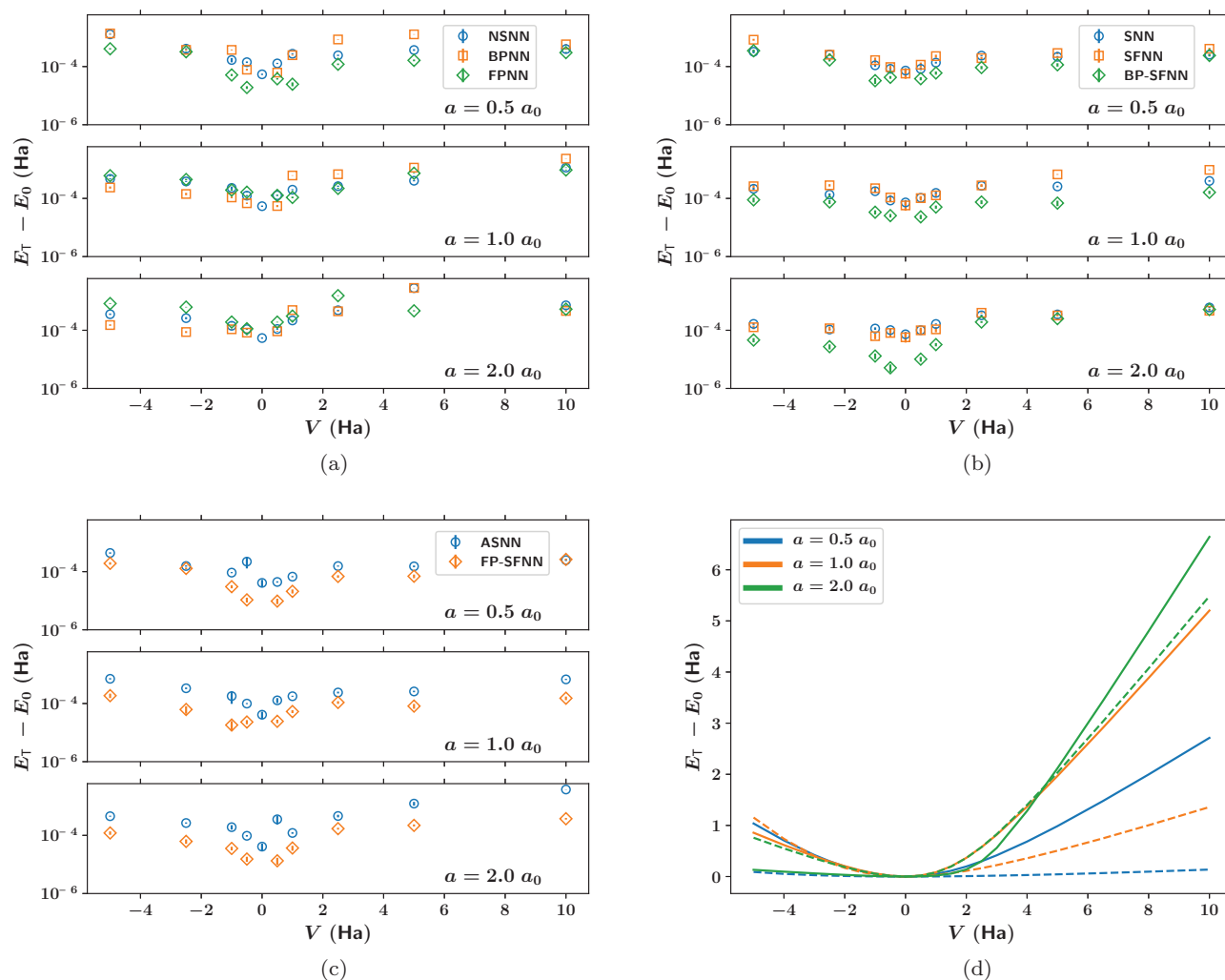


Figure 3. Energy residuals $E_T - E_0$, where E_0 is the exact solution,^[34] as obtained with a) all not strictly exchange symmetric NNWFs, that is, simple (NSNN), simple bosonic product (BPNN), and simple fermionic product (FPNN) NNWF; b) all strictly exchange symmetric NNWFs, that is, symmetrized (SNN), symmetric-featured (SFNN), and symmetric-featured bosonic product (BP-SFNN) NNWF; c) all strictly exchange antisymmetric NNWFs, that is, antisymmetrized (ASNN) and symmetric-featured fermionic product (FP-SFNN) NNWF; as well as d) bosonic (solid lines) and fermionic (dashed lines) harmonic oscillator ground states (the determinantal parts of our product NNWFs), for various potential ranges a and potential strengths V . Note the logarithmically scaled residual axis and that for product NNWFs, the trivial case of $V = 0$ is left out.

without applying an antisymmetry operator, or using special coordinates to guarantee antisymmetry of the product.

Now turning to the second group of wave functions, the energy residuals of all strictly exchange symmetric NNWFs is shown in Figure 3b, and those of all strictly exchange antisymmetric NNWFs in Figure 3c. One main observation is that our symmetric-featured NNWF performs only marginally different to either simple or symmetrized NNWFs. However, if we use a network with symmetric feature input for the Jastrow part of the bosonic product ansatz, we obtain variationally energies that are consistently similar or even lower than those of any other bosonic NNWF ansatz that we investigated. Similarly, the symmetric-featured fermionic product NNWF performs better throughout than all other fermionic NNWFs.

Therefore, we found the product-type NNWFs to offer generally superior accuracy over the simple NNWF, at least when

the imposed determinantal part of the wave function is already similar to the desired ground state. Furthermore, at least in combination with a product ansatz, we found the use of symmetrized features as neural network inputs to be beneficial, resulting in improved variational energies. In any case, symmetrized feature inputs are always advantageous by guaranteeing the desired exchange symmetry or antisymmetry (when used in fermionic product NNWFs), without the factorial scaling computational cost as with symmetrized or antisymmetrized NNWFs.

Finally, although there do exist differences in the accuracy of the various NNWF types, we find it noteworthy that although the same FFNN structure was employed in all these cases, the results are always satisfactory results without exception, even when just feeding raw coordinates to a FFNN and using the output as our wave function, which corresponds to our simple NNWF.

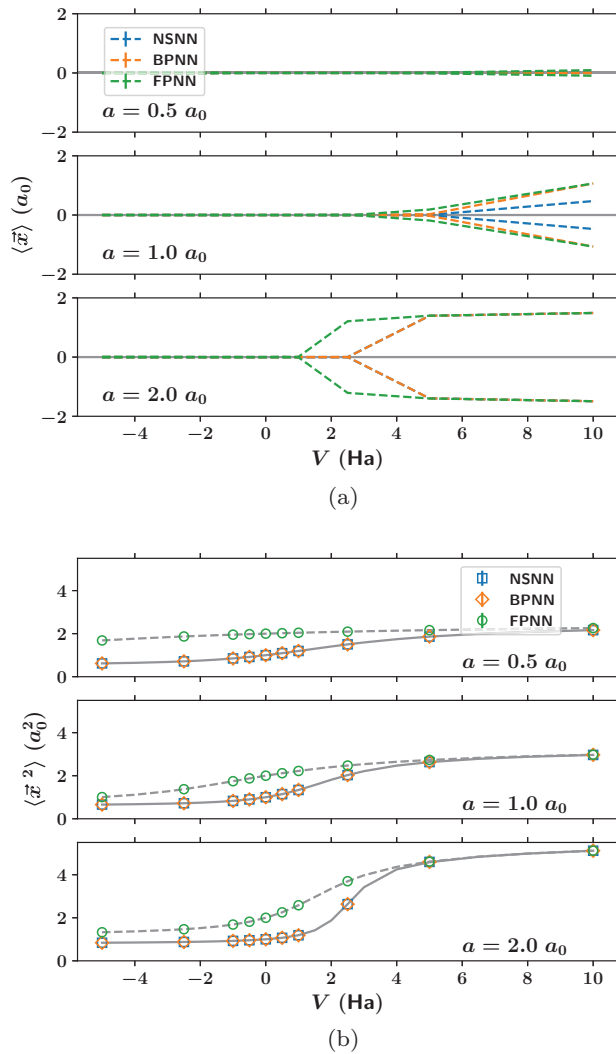


Figure 4. Position expectation values a) $\langle x_1 \rangle, \langle x_2 \rangle$ and b) $\langle \vec{x}^2 \rangle = \langle x_1^2 \rangle + \langle x_2^2 \rangle$, as obtained by our non-symmetric NNWFs, that is, simple NNWF (NSNN), simple bosonic product NNWF (BPNN), and simple fermionic product NNWF (FPNN), respectively. Grey lines depict the exact expectation values, with the solid line being for the bosonic and a dashed line for the fermionic case.

5.1.2. Asymmetry Analysis

Although we have now verified that our NNWFs approximate the desired ground state energies, we have not shown yet whether or not the quality of approximation also holds for other observables. In particular, we want to validate the correct exchange symmetry and positional observables of the non-symmetric group of NNWFs. For this purpose, we computed the positional observables $\langle \vec{x} \rangle$ and $\langle \vec{x}^2 \rangle$ (Figure 4), as well as the exchange symmetry measure (Figure 5)

$$ESM(\Psi) = \langle \Psi(x_1, x_2) | \Psi(x_2, x_1) \rangle / \langle \Psi(x) | \Psi(x) \rangle \quad (29)$$

which yields 1 for a symmetric Ψ , -1 for an antisymmetric Ψ , and 0 for a completely asymmetric Ψ . To complete the picture, we also

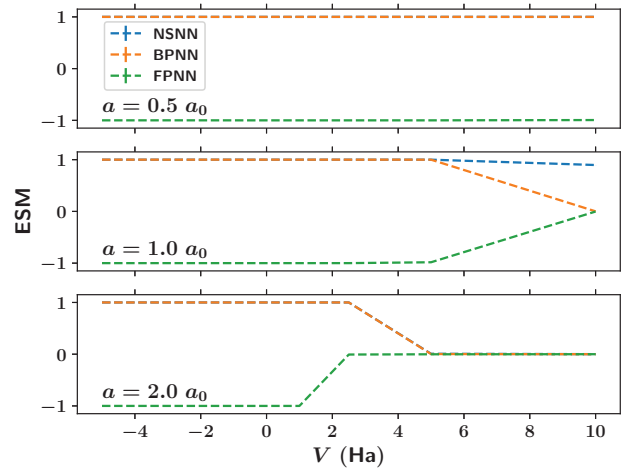


Figure 5. The ESM for all of our non-symmetric NNWFs, that is, simple NNWF (NSNN), simple bosonic product NNWF (BPNN), and simple fermionic product NNWF (FPNN), applied to the soft-core harmonic trap system. An exactly symmetric/antisymmetric wave function would yield an ESM of $+1/-1$, respectively.

computed the particle density $n(x) = \langle \sum_i \delta(x_i - x) \rangle$, as shown in Figure 6.

Looking at the results in Figures 4 and 5, respectively, it becomes immediately obvious that for certain Hamiltonian parameters $a \gg 0$ and $V \gg 0$, the observed exchange symmetry and position expectation values do not match the exact expectation values. Because we are considering indistinguishable particles and an external potential that is symmetric around the origin, we would expect both position expectation values $\langle \vec{x} \rangle$ to be identical and zero, regardless of Hamiltonian parameters. In contrast though, for the just mentioned choices for a and V , we find the non-symmetric NNWFs to predict the two particles being located on opposite sides of the trap, that is, $\langle x_1 \rangle = -\langle x_2 \rangle \neq 0$, and the symmetry measure is vanishing simultaneously. At the same time, the energy E_T and other expectation values like $\langle \vec{x}^2 \rangle$ and $n(x)$ remain close to the exact expectation. These properties indicate that our NNWF has actually learned a quasi-degenerate state of crystallized distinguishable particles, instead of the targeted bosonic or fermionic ground states. Notably, we observe this phenomenon only in a regime where bosonic and fermionic ground states itself become increasingly quasi-degenerate, which is also nicely illustrated by the density profiles in Figure 6. The behavior of our non-symmetric NNWFs for large V is fortunately not surprising, considering the analysis of the exact Hamiltonian eigenstates in the hard-core limit $V \rightarrow \infty$ ^[34] where the states can be exactly constructed from single-particle orbitals located on the left and right sides of the trap.

This is to say, we learned that if we chose to employ NNWFs without strict exchange symmetry to describe indistinguishable particles, the VMC optimization of our NNWF might prefer an asymmetric state of distinguishable particles whenever the corresponding energy is sufficiently close to the ground state energy. Hence, when for some reason using symmetric features is not an option and a non-symmetric NNWF must be used, one may check a diagnostic like Equation (29) to verify the desired symmetry, if this is of practical relevance. However, if exchange

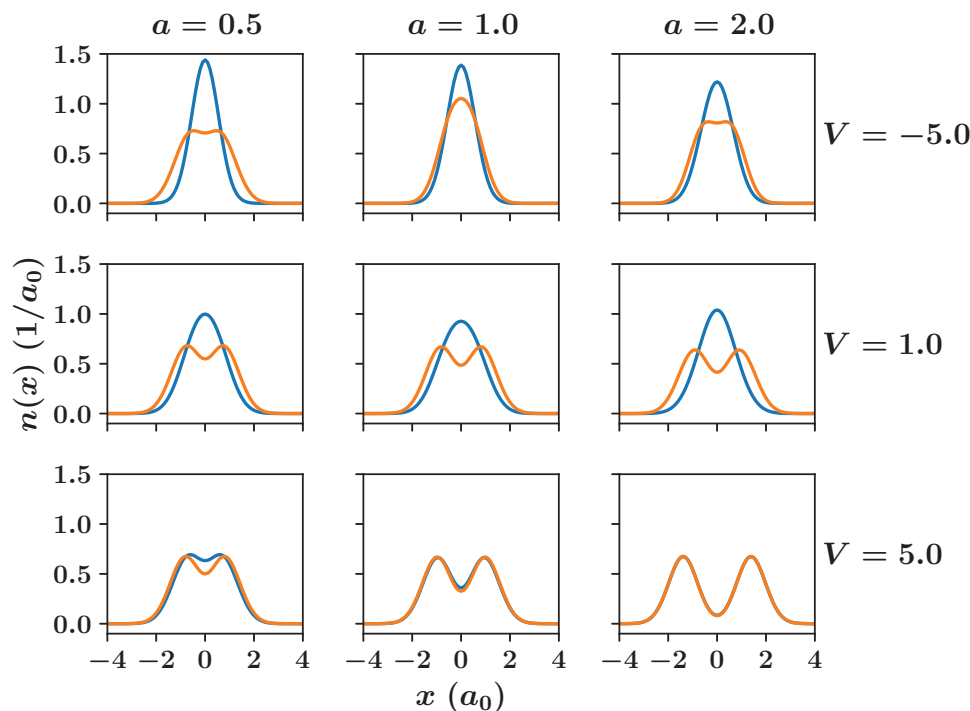


Figure 6. Density profiles $n(x)$ for the soft-core harmonic trap system, as obtained with simple bosonic (blue) and fermionic (orange) product NNWFs, respectively. All values are denoted in atomic units. The results are visually identical with the exact density profiles presented in ref. [34].

symmetry of the NNWF can be guaranteed, for example, by providing the used FFNN with an exchange symmetric input, the phenomenon discussed in this subsection is directly avoided.

5.2. Hydrogen Dimer

For the hydrogen dimer H_2 , we considered both the simple NNWF (in two different sizes), as well as our simple bosonic product NNWF (as described in Section 3.2), to compute optimized energies for a range of protonic distances $R_{HH} = |\mathbf{R}_1 - \mathbf{R}_2|$. The resulting energy curve $E_T(R_{HH})$ represents an approximation to the Born–Oppenheimer potential of the hydrogen dimer and is displayed in **Figure 7a**, with the remaining differences to the exact potential energy curve $E_0(R_{HH})$ shown in **Figure 7b**. The reference energy curve $E_0(R_{HH})$ is taken from ref. [35] and extrapolated to the complete basis set limit, where for small distances a James–Coolidge basis is used, whereas Heitler–London functions with arbitrary polynomial in electron variables are employed for large distances. Whereas the simple NNWF underestimates the exact binding energy, this can be systematically improved by increase the size of the NNWF. The bosonic product NNWF, however, quantitatively reproduces the binding energy already for rather small sizes, which demonstrates its ability to accurately recover dynamic electron correlation effects. Nevertheless, despite the differences in accuracy, all of our NNWFs correctly predict the equilibrium distance to be at $R_{HH} = 1.4011 a_0$.

However, the stretched H_2 molecule represents a rigorous challenge for even the most used accurate quantum-chemical approaches, such as two-particle reduced density matrix theory,^[37,38] natural orbital functional theory,^[39,40] and density

matrix renormalization group approaches,^[41,42] to name just a few. This well-known problem is attributed to the multi-reference character of the stretched H_2 molecule, or static electron correlation that arises in situations with degeneracy or near-degeneracy, as in transition metal chemistry and strongly correlated systems in general.^[43] As a consequence, the stretched H_2 molecule and similar problems are typically dealt with using multi-determinant wave functions,^[44,45] but for larger systems with many degeneracies, the number of determinants quickly becomes unfeasible. In fact, due to the strong multi-reference character of the stretched H_2 molecule, the potential energy is substantially overestimated upon dissociation when using the simple NNWF. On the one hand, the error can be effectively suppressed in a systematic manner by increasing the size of the FFNN within the simple NNWF, as can be seen in **Figure 7b**. On the other hand, our bosonic product NNWF is able to reproduce the exact full configuration-interaction binding curve with great accuracy. This is to say that despite the strong multi-reference character of the stretched H_2 molecule, the bosonic product NNWF is not only capable to recover the dynamic, but also the static correlation energy.

To complement the energy results, in **Figure 8**, the ESM diagnostics of Equation (29) is shown for all employed NNWFs. Whereas all of our NNWFs successfully learned approximately exchange symmetric states for $R_{HH} < 3 a_0$, they grasp asymmetric states starting from some distance $3 a_0 < R_{HH} < 6 a_0$, with the simple NNWFs becoming asymmetric earlier than the bosonic product NNWF. Just as for the soft-core harmonic trap system, the asymmetry stems from a localization of both particles on opposite sides of the system (here along the H–H axis), that is, the NNWFs adopted a state of distinguishable particles.

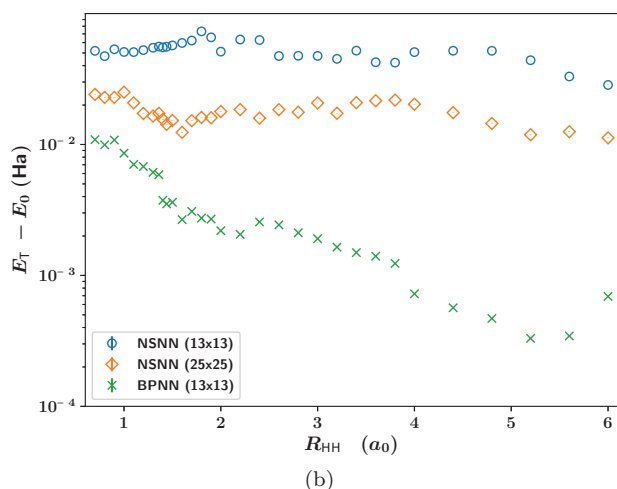
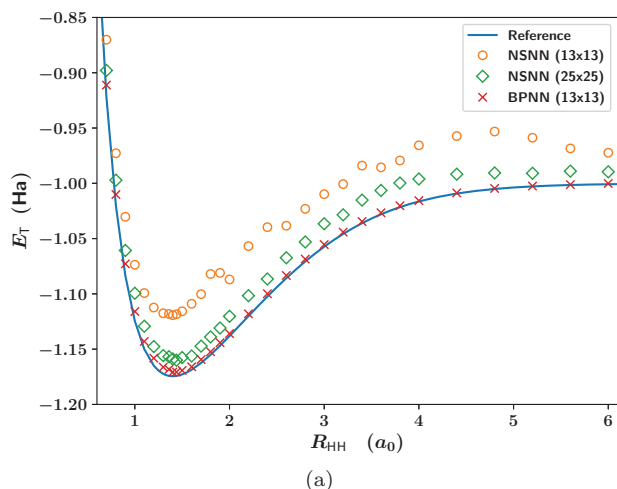


Figure 7. a) Energies and b) energy residuals of the H_2 molecule as a function of separation distance R_{HH} within the Born–Oppenheimer approximation, as obtained by our simple NNWF (NSNN) and simple bosonic product NNWF (BPNN), with 13 and 25 hidden units (including 1 offset unit) per hidden layer.

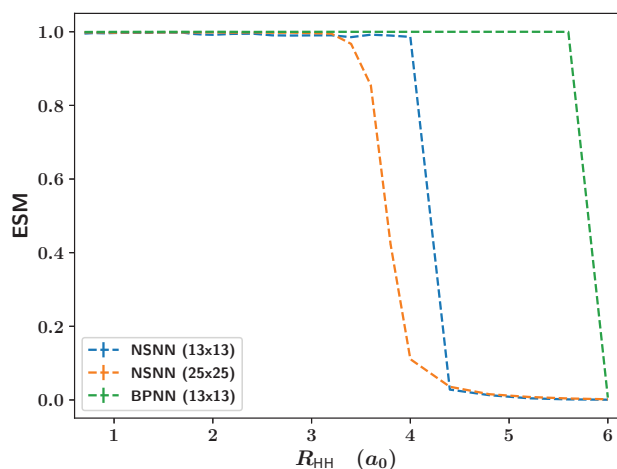


Figure 8. The ESM of our simple NNWF (NSNN) and simple bosonic product NNWF (BPNN) when employed to the H_2 molecule.

6. Conclusions

To summarize, a novel VMC method using FFNN-based trial wave functions in continuous space is presented to approximate the ground states of model Hamiltonians to (in principle) arbitrary accuracy. We have formulated several versions of such FFNN-based trial wave functions, with different properties regarding their exchange symmetry and practicality. All NNWFs were tested on a correlated, but exactly solvable, 2-particle system, the soft-core harmonic trap. In all cases, the exact ground state energies were predicted with good accuracy, in particular when considering the use of a relatively small FFNN. Nevertheless, we could show that including problem-specific knowledge within the NNWF construction offers potentially superior accuracy over a more simple, but general approach. This distinction in accuracy between the different NNWFs was more pronounced when for the H_2 molecule within the Born–Oppenheimer approximation.

We observed that NNWFs without guaranteed exchange symmetry were prone to learn states of distinguishable particles whenever such a state was energetically close to the ground state of indistinguishable particles, though the practical consequences of this behavior remains indeterminate. Research on novel methods for providing exchange symmetric input to the FFNN, thereby preventing the asymmetry inherently, are currently underway and will be reported elsewhere.

In conclusion, we have demonstrated that our NNWFs represent a flexible trial wave function that can be straightforwardly applied to a variety of different Hamiltonians without necessarily requiring problem-specific adjustments. We expect to achieve a significant speed-up of the optimization process by employing more suitable differentiation techniques to compute the necessary gradients, in particular by employing a reverse-mode differentiation scheme. However, our currently used forward-accumulation scheme will most likely remain an efficient choice for all second-order input derivatives, which are required to compute the exact kinetic energy of the NNWF.

Finally, it is worth noticing that the (anti-)symmetry requirement of the trial wave function can be considered as a serious limitation of our method, since the straightforward and general way of constructing (anti-)symmetric NNWFs via (anti-)symmetrization operators (see Equations (10b) and (11b)) is computationally not feasible for many-body applications. Fortunately, the fact that the lowest energy state is always bosonic makes it possible to simulate bosons by relying on the fact that the optimization will make the NNWF approximately symmetric. When one is interested in the simulation of fermions, however, the problem becomes more serious. In fact, the approach we used in this work corresponds to a fixed-node approximation.^[46] This means that the quality of our solution will depend on the nodal surface resulting from the choice of the functions that form the Slater determinant. The possibility to go beyond the fixed-node approximation and recover even more static electron correlation using a single Slater determinant only by means of the backflow transformation,^[17,18,25,47] shadow wave function,^[5,6,48–50] or resonating valence bond approach,^[51–53] is to be underlined and will be presented elsewhere.

Appendix

A.1. Activation Functions

To investigate the sensitivity of the chosen activation function on the eventually achieved accuracy, besides the a_T and a_G hidden layer activation functions mentioned in Section 2.2, we also explored the identity function for some of the hidden units, that is,

$$a_I(p) = p \quad (\text{A1})$$

Furthermore, in addition to an exponential activation function for the output unit, we also tried the commonly employed logistic activation function

$$a_L(p) = \frac{1}{1 + \exp(-p)} \quad (\text{A2})$$

which is bounded between 0 and 1. Note that this range restriction is not a problem in our VMC method, as the scale or normalization of the wave function is not relevant.

We dismissed, however, the common choice of linear output activation functions, since in our testing we observed unfavorably larger oscillations around zero for input values, where the wave function has a small amplitude. Moreover, in the bosonic case, we want to avoid a wave function with variable sign anyway.

The energy residuals results for the harmonic soft-core trap together with selected combinations of hidden layer and output activation functions are displayed in **Figure A1**. The corresponding results for the combination of a_T and a_G hidden layer activation function with either a_L or a_E on the output, respectively, are shown in **Figure A2**.

A.2. Hidden Layer Structure

Similarly, we also investigated how the accuracy changes when changing the size and number of hidden layers. An overview of our results is shown in **Figure A3**.

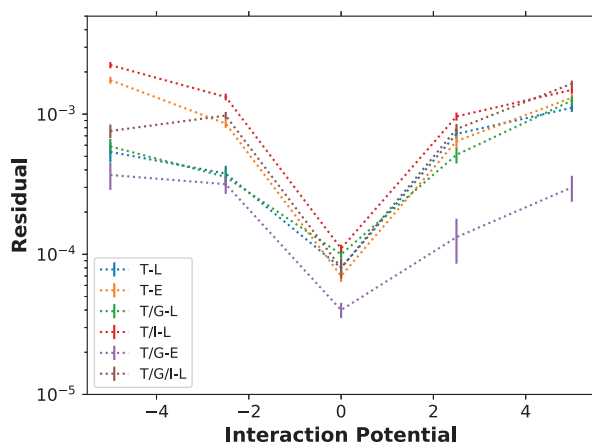


Figure A1. The energy residuals with respect to the exact ground state energies for the simple NNWF applied to the bosonic harmonic soft-core trap system. Results for different combinations of hidden layer and output activation functions are displayed, where T refers to hyperbolic tangent sigmoid, G to Gaussian, I to the identity, L to logistic, and E to exponential activation functions, respectively.

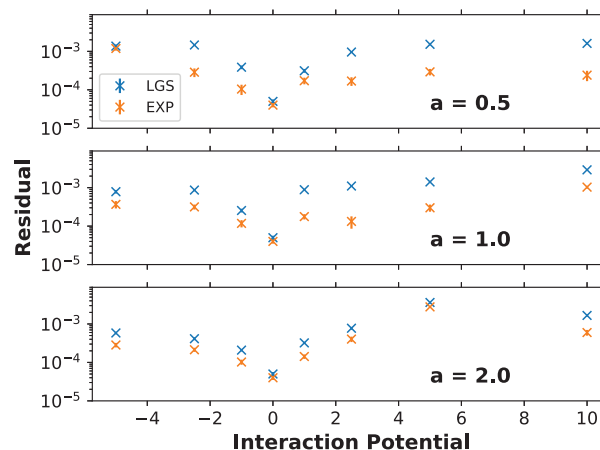


Figure A2. The energy residuals with respect to the exact ground state energies for the simple NNWF applied to the bosonic harmonic soft-core trap system. Results for T/G-L (LGS) and T/G-E (EXP) as hidden layer and activation functions are shown for different Hamiltonian parameters, where T refers to hyperbolic tangent sigmoid, G to Gaussian, L to logistic, and E to exponential activation functions, respectively.

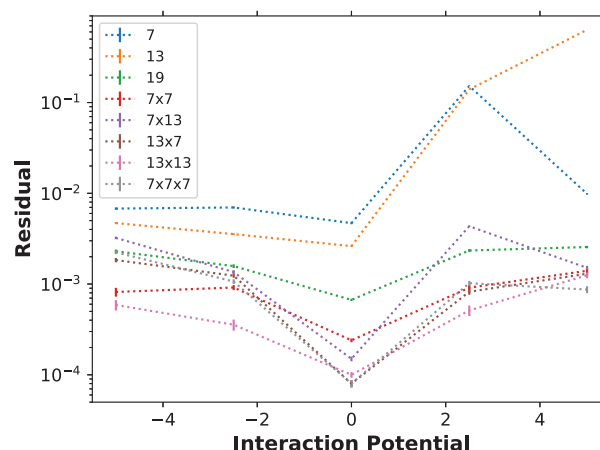


Figure A3. The energy residuals with respect to the exact ground state energies for the simple NNWF applied to the bosonic harmonic soft-core trap system. Results for the T/G-E activation function configuration is shown as a function of different number of hidden layers and hidden units, where T refers to hyperbolic tangent sigmoid, G to Gaussian, and E to exponential activation functions, respectively.

Acknowledgements

The generous allocation of computing time on the FPGA-based supercomputer “Noctua” by the Paderborn Center for Parallel Computing (PC²) is kindly acknowledged. This project received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No 716142).

Correction added on 14 April 2021, after first online publication: Projekt Deal funding statement has been added.

Open access funding enabled and organized by Projekt DEAL.

Conflict of Interest

The authors declare no conflict of interest.

Data Availability Statement

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Keywords

Monte Carlo simulations, quantum Monte Carlo simulations, variational Monte Carlo simulations

Received: October 27, 2020

Revised: January 5, 2021

Published online: January 25, 2021

- [1] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller, *J. Chem. Phys.* **1953**, 21, 1087.
- [2] W. L. McMillan, *Phys. Rev.* **1965**, 138, A442.
- [3] D. M. Ceperley, B. J. Alder, *Science* **1986**, 231, 555.
- [4] W. M. C. Foulkes, L. Mitars, R. J. Needs, G. Rajagopal, *Rev. Mod. Phys.* **2001**, 73, 33.
- [5] F. Calcevaccia, F. Pederiva, M. H. Kalos, T. D. Kühne, *Phys. Rev. E* **2014**, 90, 053304.
- [6] F. Calcevaccia, T. D. Kühne, *Z. Naturforsch. A* **2018**, 73, 845.
- [7] K. Hornik, *Neural Networks* **1991**, 4, 251.
- [8] Y. LeCun, Y. Bengio, G. Hinton, *Nature* **2015**, 521, 436.
- [9] I. E. Lagaris, A. Likas, D. I. Fotiadis, *Comput. Phys. Commun.* **1997**, 104, 1.
- [10] G. Carleo, M. Troyer, *Science* **2017**, 355, 602.
- [11] X. Gao, L.-M. Duan, *Nat. Commun.* **2017**, 8, 662.
- [12] H. Saito, M. Kato, *J. Phys. Soc. Jpn.* **2017**, 87, 014001.
- [13] I. Glasser, N. Pancotti, M. August, I. D. Rodriguez, J. I. Cirac, *Phys. Rev. X* **2018**, 8, 011006.
- [14] Z. Cai, J. Liu, *Phys. Rev. B* **2018**, 97, 035116.
- [15] Y. Levine, O. Sharir, N. Cohen, A. Shashua, *Phys. Rev. Lett.* **2019**, 122, 065301.
- [16] G. Carleo, Y. Nomura, M. Imada, *Nat. Commun.* **2018**, 9, 5322.
- [17] M. Ruggeri, S. Moroni, M. Holzmann, *Phys. Rev. Lett.* **2018**, 120, 205302.
- [18] D. Luo, B. K. Clark, *Phys. Rev. Lett.* **2019**, 122, 226401.
- [19] J. Kessler, F. Calcevaccia, T. D. Kühne, arXiv:1904.10251 [physics.comp-ph].
- [20] D. Pfau, J. S. Spencer, A. G. D. G. Matthews, W. M. C. Foulkes, *Phys. Rev. Res.* **2020**, 2, 033429.
- [21] J. Hermann, Z. Schätzle, F. Noe, *Nat. Chem.* **2020**, 12, 891.
- [22] W. K. Hastings, *Biometrika* **1970**, 57, 97.
- [23] Y. Nagaoka, T. Usui, *Prog. Theor. Phys. Suppl.* **1968**, E68, 392.
- [24] Y. Kwon, D. M. Ceperley, R. M. Martin, *Phys. Rev. B* **1998**, 58, 6800.
- [25] M. Holzmann, D. M. Ceperley, C. Pierleoni, K. Esler, *Phys. Rev. E* **2003**, 68, 046707.
- [26] P. López Ríos, A. Ma, N. D. Drummond, M. D. Towler, R. J. Needs, *Phys. Rev. E* **2006**, 74, 066701.
- [27] S. Azadi, W. M. C. Foulkes, T. D. Kühne, *New Journal of Physics* **2013**, 15, 113005.
- [28] M. Taddei, M. Ruggeri, S. Moroni, M. Holzmann, *Phys. Rev. B* **2015**, 91, 115106.
- [29] S. Azadi, T. D. Kühne, *J. Chem. Phys.* **2017**, 146, 084503.
- [30] S. Azadi, T. D. Kühne, *Phys. Rev. B* **2018**, 97, 205428.
- [31] J. Behler, *J. Chem. Phys.* **2011**, 134, 074106.
- [32] H. Eshet, R. Z. Khaliullin, T. D. Kühne, J. Behler, M. Parrinello, *Phys. Rev. B* **2010**, 81, 184107.
- [33] A. P. Bartók, R. Kondor, G. Csányi, *Phys. Rev. B* **2013**, 87, 184115.
- [34] P. Kościuk, T. Sowiński, *Sci. Rep.* **2018**, 8, 48.
- [35] K. Pachucki, *Phys. Rev. A* **2010**, 82, 032509.
- [36] D. P. Kingma, J. Ba, arXiv:1412.6980 [cs] **2014**.
- [37] D. A. Mazziotti, *Phys. Rev. Lett.* **2004**, 93, 213001.
- [38] D. A. Mazziotti, *Phys. Rev. Lett.* **2008**, 101, 253002.
- [39] S. Goedecker, C. J. Umrigar, *Phys. Rev. Lett.* **1998**, 81, 866.
- [40] R. Gebauer, M. H. Cohen, R. Car, *Proc. Nat. Acad. Sci. USA* **2016**, 113, 12913.
- [41] G. Chan, M. Kallay, J. Gauss, *J. Chem. Phys.* **2004**, 121, 6110.
- [42] J. Hachmann, W. Cardoen, G. Chan, *J. Chem. Phys.* **2006**, 125, 144101.
- [43] A. J. Cohen, P. Mori-Sanchez, W. Yang, *Science* **2008**, 321, 792.
- [44] R. J. Bartlett, M. Musiał, *Rev. Mod. Phys.* **2007**, 79, 291.
- [45] A. Köhn, M. Hanauer, L. A. Mück, T. Jagau, J. Gauss, *WIREs Comput. Mol. Sci.* **2013**, 3, 176.
- [46] D. M. Ceperley, B. J. Alder, *Phys. Rev. Lett.* **1980**, 45, 566.
- [47] R. P. Feynman, M. Cohen, *Phys. Rev.* **1956**, 102, 1189.
- [48] S. Vitiello, K. Runge, M. H. Kalos, *Phys. Rev. Lett.* **1988**, 60, 1970.
- [49] L. Reatto, G. L. Masserini, *Phys. Rev. B* **1988**, 38, 4516.
- [50] F. Calcevaccia, T. D. Kühne, *Eur. Phys. Lett.* **2015**, 110, 20011.
- [51] P. W. Anderson, *Science* **1987**, 235, 1196.
- [52] S. Azadi, R. Singh, T. D. Kühne, *Int. J. Quantum Chem.* **2015**, 115, 1673.
- [53] K. Nakano, C. Attaccalite, M. Barborini, L. Capriotti, M. Casula, E. Coccia, M. Dagrada, C. Genovese, Y. Luo, G. Mazzola, A. Zen, S. Sorella, *J. Chem. Phys.* **2020**, 152, 204121.