Inspired by the universal approximation theorem and widespread adoption of artificial neural network techniques in a diversity of fields, we propose feed-forward neural networks as general purpose trial wave functions for quantum monte carlo (QMC) simulations of many-body systems. To verify their practical success, we employ various realizations of this ansatz in QMC simulations of an exactly solvable model system of two trapped interacting particles. Finally, we use the same technique to accurately predict the binding curve of the hydrogen molecule, from first principles.

I. Introduction

Although the stationary Schrödinger equation describes the most basic form of quantum mechanics, finding accurate eigenstates and eigenvalues of the equation remains a hard problem when realistic physical systems are concerned. Especially the important contribution of the so-called correlation energy requires expensive computations when high accuracy is desired, which typically exhibit high-order polynomial or even exponential scaling of effort with number of particles, rendering application to larger systems of particles infeasible.

One class of methods for solving the many-body Schrödinger equation efficiently are the so-called variational Monte Carlo (VMC) methods [1], which rely on optimizing parametrized trial wave functions according to the Rayleigh-Ritz variational principle and evaluating the apparent high-dimensional integrals by the means of Monte Carlo (MC) integration. Typically, the approach involves the postulation of problem-specific trial wave function formulations, in an attempt to reduce the number of required variational parameters by making use of physical intuition. The apparent downside of such an ansatz are the need for physical intuition and the inflexibility of a specialized trial wave function, which is particularly problematic when, for example, phase transitions are meant to be investigated.

In this work we want to consider an overall similar, but conceptionally different approach. Instead of crafting specific trial wave functions, we want to investigate a novel class of very general trial wave functions built from Artificial Neural Networks (ANN) and in particular Feedforward Neural Networks (FFNN). This idea is fueled by the Universal Approximation Theorem, which states that FFNNs can approximate any continuous function on a finite measure to arbitrary accuracy [2]. Furthermore, ANNs already prove to be a successful “black-box” approach for a variety of problems in different fields and they are seen as well-suited for dealing with high-dimensional input [3]. In turn however, the number of variational parameters within FFNNs grows large quickly and some need for intuition is moved to the question of how particle coordinates are presented to the FFNN and what activation functions are to be used in the network.

The (to our knowledge) first successful application of ANNs as wave functions for simple one-, two- and three-dimensional systems was published already about 20 years ago [4], but the bulk of work in this direction is very recent (2017-2018) [5-12]. However, none of these works consider the use of FFNNs within a VMC framework in continuous space, so the method that we present can be considered novel.

In summary, we aim to present a new method for approximating the ground states of Schrödinger equations in continuous space, by making use of FFNNs in our trial wave functions. To train the employed neural networks, we devise a gradient-based VMC optimization scheme. Both components of the method promise beneficial scaling of computational effort with dimensionality and allow for massively parallel execution, providing efficient use of modern supercomputers. Finally, no input of any existing data or knowledge is required in our approach, rendering it a true ab-initio method.

II. Methods

In this section we want to present our novel method for approximating ground state wave functions of continuous quantum systems by employing basic FFNNs as trial wave functions.

The first cornerstone of our method is the Rayleigh-Ritz variational principle, justifying a systematic gradient-based strategy to train the employed neural net-
works. We compute the apparent integrals by the means of importance sampled MC integration, resulting in a VMC optimization scheme. This shall be explained in more detail in the first following subsection.

Subsequently, we describe how the specific neural network wave functions (NNWF) 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 versions.

Finally, we explain the gradient-based stochastic optimization scheme used for minimizing the energy of our NNWFs.

II.1. Variational Monte Carlo

From the non-relativistic time-independent spatial Schrödinger equation

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

we know that the exact spectrum $\Psi_{\alpha}(R)$ 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 $E_{\alpha}$ are the energies of the respective states. The state with the lowest energy $E_0$ is called ground state and labeled as $\Psi_0(R)$.

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

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

where $\Psi_T = \Psi_T(R; \beta)$ is a parametrized trial wave function with parameters $\beta$ 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 $\Psi_T$ by varying $\beta$ so as to minimize $E_T$.

The trial wave functions $\Psi_T(R)$ are functions 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 Eq. 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 [13]. In fact, given the trial wave function $\Psi_T(R)$, it is sufficient to compute the

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

FIG. 1. Illustration of a feedforward NNWF with two input coordinates, two hidden layers, $4 + 1$ hidden units each and exponential output activation function. Input units are depicted as red circles, offset units are orange, hidden units blue and output units purple. Graphs inside of the circles represent the activation functions. The grey lines depict the unidirectional links between units, transferring information from left to right.

local energy

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

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 Eq. 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. To optimize the parameters of the trial wave function, we use a gradient-based stochastic optimization algorithm, described in more detail in subsection II.3.

II.2. Feedforward Neural Network Wave Functions

In the following, we describe how we construct trial wave functions for VMC simulations from basic FFNNs.

The structure of such a network is illustrated in Fig. 1. The employed neural networks 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 $p(x; \beta)$ 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, i.e.

$$p(x; \beta) = \beta \cdot x = \sum_i \beta_i x_i,$$

where $\beta_i$ 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:

- hyperbolic tangent sigmoid: $a_T(p) = \frac{2}{1+\exp(-2p)} - 1$
- Gaussian: $a_G(p) = \exp(-p^2)$

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

$$a_E(p) = \exp(p),$$

which in general appears to be rarely used in ANNs. But, inspired by the omnipresence of exponentials in wave function formulations, we considered it for our use case. Using this activation function on the output unit means that the FFNN has strictly positive output values, which will be relevant in some of the following discussions.

For informations about results with different activation function choices, we refer to the appendix.

### II.2.1. Simple NNWF

For a first straightforward NNWF approach, we can feed the raw particle coordinates $\mathbf{R}$ directly into the described network via the input layer and employ the network’s output 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 wave function as

$$\Psi_{NS}(\mathbf{R}) =$$

$$a_o \left( \beta_{o,0} + \sum_{i=1}^{n_h} \beta_{o,i} a_{h,i} \left( \beta_{h,i,0} + \sum_{j=1}^{D+N} \beta_{h,i,j} R_j \right) \right),$$

where $a_o$ is the chosen output activation function ($a_L$ or $a_E$) and $a_{h,i}$ the selected hidden layer activation functions ($a_T$ and/or $a_G$). The corresponding connection weights within the output and hidden units, which will be determined by the training process, are denoted as $\beta_o$ and $\beta_{h,i}$, 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 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 needs 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 [14]. In this case, by minimizing the energy, we would expect an asymmetric NNWF to approximate the exchange-symmetric ground state, thus becoming an approximately symmetric wave function.

Although this ansatz may seem a bit crude, we still consider it interesting for a 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. In the following, we will refer to it as “simple NNWF”.

#### II.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 unlikely if an approximation to an antisymmetric ground state is needed. Typically, the antisymmetric ground state is of higher energy than the corresponding symmetric ground state, so minimizing the energy of an asymmetric NNWF is expected to yield an approximately symmetric state. Moreover, in the special case of strictly positive valued output activation functions, the simple 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

\[ \Psi_S(R) = S \Psi_{NS}(R) \quad \Psi_{AS}(R) = A \Psi_{NS}(R), \]

respectively, where

\[ S = \frac{1}{N!} \sum_{p \in S_N} \hat{P} \quad A = \frac{1}{N!} \sum_{p \in S_N} (-1)^\pi \hat{P}, \]

with \( S_N \) being the symmetric group of permutations \( \hat{P} \)
and \( \pi \) denoting the parity of the individual permutation.

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

\[ \Psi_{S/AS}(R) = \frac{1}{2} (\Psi_{NS}(R_1, R_2) \pm \Psi_{NS}(R_2, R_1)). \]

Constructing the NNWF in this way guarantees the
desired computational scaling. 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.

II.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 merits,
we would not want to stay away from more specialized
variants, at least when they prove to be easy to realize
and more efficient. One possible idea in this direction is
to utilize a NNWF \( \Psi_{NN}(R) \) merely as a modulating factor for an imposed problem-specific base wave function
\( \Phi(R) \), i.e.

\[ \Psi_P(R) = \Psi_{NN}(R) \Phi(R). \]

The base wave function \( \Phi(R) \) 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, \( \Phi(R) \) can be used to directly define the sign
of the full product \( \Psi_P(R) \), given that a strictly positive
valued \( \Psi_{NN}(R) \) is used. Then \( \Psi_P(R) \) has the same sign
as \( \Phi(R) \) for all \( R \) and is equal to zero only at the same
nodal surface \( \{ R : \Phi(R) = 0 \} \). This property effectively
restricts the set of wave functions that such a NNWF can accurately approximate, at least if \( \Phi(R) \) actually has
nodes.

To approximate bosonic ground states, we consider
a symmetric product of identical single-particle orbitals
\( \phi(r) \) as a base wave function, leading to

\[ \Psi_{BP}(R) = \Psi_{NN}(R) \phi(r_1) \phi(r_2) \ldots, \]

where \( r_i \) depict the coordinates of all individual particles.
We will refer to \( \Psi_{BP}(R) \) as “bosonic product NNWF”.

For fermionic ground states, however, we use an
antisymmetric Slater-determinant \( \Phi_{SD} \) of given single-particle orbitals \( \phi_i(r) \) as the base wave function:

\[ \Psi_{FP}(R) = \Psi_{NN}(R) \Phi_{SD}(R) \]

We denote \( \Psi_{FP}(R) \) as “fermionic product NNWF”. As
already alluded to above, given a strictly positive \( \Psi_{NN} \),
the nodes of \( \Psi_{FP}(R) \) are defined by \( \Phi_{SD}(R) \). This is
to say that as long as \( \Phi_{SD}(R) \) does not change during
the optimization, the nodes are fixed. If the nodal
surface should be variable instead, the considered orbitals
need to contain variational parameters to be optimized
together with the FFNN weights. One known possibility
in this direction is the use of backflow orbitals [12].
However, for the applications in this work, we are able
to rely on a fixed node ansatz.

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

II.2.4. Symmetric-Featured NNWF

Another possible optimization approach concerns the way
information is fed to the actual FFNN. Until now we simply used all \( D \times N \) raw particle coordinates \( R \) as input, but this choice might not be the ideal one in all cases.

For example considering exchange-symmetry, it is easy
to see that the raw coordinates contain some redundant
information, because for the wave function it doesn’t matter if particle 1 is at \( x_1 \) and particle 2 at \( x_2 \) or if it is the other way around. Using a symmetric representation of particle coordinates instead would guarantee exchange symmetry of the NNWF and potentially increase accuracy. And in general, an inspection of the specific Hamiltonian might lead to more efficient representations of particle coordinates.

We implement the possibility of such representations by replacing the \( D \times N \) particle coordinates \( R_j \) in Eq. 7
with \( n_f \) functions \( f_j(R) \) (“features”), resulting in

\[ \Psi_{SF}(R) = \sum_{i=1}^{n_h} \beta_{o,i} a_{h,i} \left( \sum_{j=1}^{n_f} \beta_{h,i,j} f_j(R) \right) \]

for a network with only a single hidden layer. In this work, we want to restrict us to exchange-symmetric
This means that in our MC integration scheme we can compute the gradients $G_i$ of our cost function $C$ as

$$G_{MC,i} = \frac{2}{N_{MC}} \sum_{i=1}^{N_{MC}} \left( E_{\text{loc}} \frac{\partial \phi}{\partial \psi} - E_{MC} \sum_{j=1}^{n_{MC}} \frac{\partial \phi}{\partial \psi} \right) + 2 \frac{\lambda r}{n_{\beta}} \beta_i,$$

(17)

where the summains of sums $\sum_{i=1}^{N_{MC}}$ are evaluated along the importance sampled random walk, as discussed in section II.3.

In conclusion, we have devised a gradient-based optimization scheme to minimize the energy of FFNN-based trial wave functions for a given Hamiltonian. The variational principle guarantees that this energy-minimization scheme is a valid strategy to approximate the ground state wave function of the system.

III. Applications

III.1. 2-particle harmonic trap with soft-core interaction

For a first application of our NNWFs on a toy model, we selected a system of two particles in an one-dimensional harmonic potential, interacting via a potential of finite-range and constant-value. Such a system can be described by the Hamiltonian

$$\hat{H} = -\frac{\partial^2}{\partial x_1^2} + \frac{\partial^2}{\partial x_2^2} + \frac{x_1^2 + x_2^2}{2} + V(x_1 - x_2),$$

(18)

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

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

(19)

This model is a strongly correlated system, but still it was shown to be exactly solvable for both bosonic and fermionic symmetry [10]. The recipe for exact solution allowed us to obtain the exact energy eigenvalues and eigenstates for validation of our simulation results. The soft-core interaction between particles can be either attractive or repulsive and both potential range and strength can be chosen freely, providing us a rich testing environment for our NNWFs.

While the Hamiltonian [18] is all that is required to begin VMC optimization of the simple $\Psi_{NS}(x)$ or (anti-)symmetrized $\Psi_{S/A}(x)$, for the product formulations $\Psi_{BF}(x)$ and $\Psi_{FP}(x)$ we need to propose single-particle orbitals. 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 $\psi_n$. Relevant for us are the first two orbitals

$$\psi_0(x) = \pi^{-\frac{1}{4}} e^{-\frac{1}{2}x^2}, \quad \psi_1(x) = \sqrt{2} \pi^{-\frac{3}{4}} x e^{-\frac{1}{2}x^2}.$$  

(20)
From the first or both of these orbitals we build the right sides of bosonic or fermionic product NNWF as in Eq. 12 or 13 respectively. In the case of symmetric-featured NNWFs we employ

\[ f_1 = x_1^2 + x_2^2, \quad f_2 = (x_1 - x_2)^2 \]  

(21)
as input to the FFNN, instead of raw coordinates \( x \). For the given Hamiltonian, all relevant information are retained in this representation, it guarantees bosonic exchange symmetry, spatial symmetry corresponding to the Hamiltonian and it is mathematically smooth. We use this representation both for symmetric-featured simple NNWFs and for the left side of symmetric-featured bosonic/fermionic product NNWFs.

### III.2. \( H_2 \) Molecule

As a more realistic application we study the hydrogen molecule \( H_2 \) in Born-Oppenheimer approximation, i.e. we consider the electronic Hamiltonian (in atomic units)

\[
\hat{H} = -\frac{1}{2} \left( \nabla_{r_1}^2 + \nabla_{r_2}^2 \right) - \sum_{i,j=1}^2 \frac{1}{|r_i - R_j|} + \frac{1}{|r_1 - r_2|} + \frac{1}{|R_1 - R_2|},
\]

(22)

where \( r \) denote the variable electronic coordinates and \( R \) the static protonic coordinates.

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

\[
\phi(r) = e^{-|r-R_1|} + e^{-|r-R_2|},
\]

(23)

where \( r \) are the single electron coordinates and \( R_i \) the two protonic coordinates.

Unlike for the previously discussed trap, the exact ground state of the hydrogen molecule is not known. Hence, as a reference we used energies computed by full-configuration-interaction [17], which can be considered exact for our purposes. Also unlike for the trap system, we don’t devise symmetric feature coordinates, because although it is straightforward to do for the simple hydrogen molecule, we think it is more interesting to revisit the matter in future research on larger systems.

### IV. Computational Details

We used the following basic scheme for all reported simulations:

1. Least-square fit the FFNN to a first-guess solution
2. Repeatedly use Adam-VMC scheme to optimize NNWF iteratively
3. Identify the result with lowest upper energy confidence bound

The first step initializes the NNWF to a well-behaved state before the actual VMC optimization is performed, using relatively fast least-squares fitting. This procedure is not strictly necessary, but 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, we used a product of Gaussian single-particle orbitals 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 we skipped the first step and started with weights initialized to small random numbers. Note that for technical reasons we didn’t use any symmetrizing operators during fitting, even when they were used in the following VMC-optimization.

Starting with the pre-fitted FFNN from step 1 employed as NNWF, in step 2 we executed the previously described Adam optimization, with parameters \( \alpha = 0.002, \beta_1 = 0.9, \beta_2 = 0.9, \epsilon = 10^{-8}, \lambda_r = 0.005. \) The variational parameters of the NNWF were updated according to Adam rule until the last 100 optimization steps yielded constant energy, with respect to integration error. With the resulting averaged variational parameters, a final MC evaluation of the respective energy was carried out. Afterwards, the Adam-VMC procedure was restarted, beginning with the final result from the previous optimization run. This scheme was repeated until reaching a consistently fixed runtime limit of 8 hours for the trap system (16h for \( \Psi_S \) and \( \Psi_{AS} \)) and 24h for \( H_2 \) (48h for the NNWF with 25x25 hidden units).

Finally in step 3, we chose out of the individual energy results of all chained optimization runs the one with the lowest upper energy confidence bound

\[
E_{ub} = E_{MC} + 2 \cdot \sigma_{MC},
\]

(24)

with \( \sigma_{MC} \) being the integration error estimation. We do this selection, because the last optimization run is not necessarily the one with the best energy, due to the stochastic nature of the optimization. The best NNWFs selected in step 3 were then employed in extensive MC-sampling runs to evaluate the energies and other observables reported in this paper.

The calculations of energy and gradient at each optimization step were carried out via \( 4 \times 10^5 \) MC iterations, the final energy evaluation of each chained optimization run via \( 8 \times 10^7 \) iterations and the concluding observable sampling runs via \( 10^8 \) iterations, distributed among 16 CPU cores in all cases.

For the MC sampling we used all-particle moves proposed from a uniform distribution, with a maximal step size calibrated for \( \approx 50\% \) acceptance probability during 2500 equilibration steps per thread, following a random walker initialization before every MC integration.
For better stability when using NNWFs, we confined the walkers to a periodic interval or box with edge length \(-10 < x < 10\) Bohr, large enough to still allow proper simulation of the non-periodic systems of interest.

All FFNNs used to obtain our application results 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 Fig. 11 (appendix).

To realize our NNWF optimization in practice, we wrote our own C++ libraries for all involved tasks. All relevant libraries are linked in the README of the DCM-UPB/NNVMC repository on GitHub (https://github.com/DCM-UPB/NNVMC, not public yet!). Access to our actual application programs and/or data can be provided on request.

V. Results

V.1. 2-particle harmonic trap with soft-core interaction

Using the previously described optimization scheme, we applied all of our NNWF formulations to the soft-core harmonic trap system, for various choices of Hamiltonian parameters \(a\) (potential range) and \(V\) (potential strength). Exemplary, Fig. 2 shows the resulting energy values for simple bosonic and fermionic product NNWFs, in comparison with the exact bosonic and fermionic ground state energies. 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, so we refrain from displaying them in the same way.

For a more in-depth analysis of the results, we split our NNWF formulations into two groups:

1. Asymmetric NNWFs without strict exchange-symmetry, i.e. simple NNWF and simple bosonic and fermionic product NNWFs.

2. (Anti-)Symmetric NNWFs with strict bosonic or fermionic exchange-symmetry, i.e. symmetrized and antisymmetrized NNWFs, symmetric-featured NNWF and symmetric-featured bosonic and fermionic product NNWFs.

For both groups we will display all resulting energy residuals \(E_T - E_0\), i.e. the difference between VMC energy result \(E_T\) and true 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.

V.1.1. Energy residuals

Beginning with the group of asymmetric NNWFs, we display their energy residuals in Fig. 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 right side of our product NNWFs is already exactly the ground state of that system.

In Fig. 3a we are mainly looking for effects of using the simple bosonic-product NNWF (NS-BP) over the simple NNWF (NS). Overall, it appears that there is no clearly superior ansatz among the two. However, the product ansatz yields consistently similar or lower energies whenever the soft-core potential is attractive or low by absolute value. In turn it yields similar or higher energies for strong repulsive interaction. This behavior can be understood as a verification of our assumption that the product-type NNWF perform better when their imposed right side is more similar to the true ground state. For our bosonic product NNWF we are using the bosonic...
Looking at the simple fermionic product NNWF (NS-FP) 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 interaction. Especially noteworthy is that the simple fermionic product NNWF predicted accurate (residual \( < 10^{-3} Ha \)) fermionic energies without applying an antisymmetry operator or using special coordinates to guarantee antisymmetry of the product.

Now turning to the second group of wave functions, we display the energy residuals of all strictly exchange-symmetric NNWFs in Fig. 3a and those of all strictly exchange-antisymmetric NNWFs in Fig. 3c. One main observation is that our symmetric-featured NNWF (SF) performs only marginally different to either simple (NS) or symmetrized (Sym) NNWFs. However, if we use a network with symmetric feature input for the left side of the bosonic product ansatz (SF-BP), we get energy results that are consistently similar or better than any other bosonic NNWF ansatz that we used. Similarly, the symmetric-featured fermionic product NNWF (SF-FP) performs better than the other fermionic NNWFs (NS-FP and FSym).

Concluding, we found the product-type NNWFs to of-
fer superior accuracy over the simple NNWF, at least when the imposed base 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 input beneficial for resulting energies. In any case, symmetrized feature input is beneficial by guaranteeing the desired exchange-symmetry or antisymmetry (when used in fermionic product), without the factorial scaling of computational cost as with symmetrized or antisymmetrized NNWFs.

And finally, although there do exist differences in energy accuracy for the various NNWF types, we find it noteworthy that the same FFNN structure was employed in all these cases and yielded satisfactory results without exception, even when all we did was just feeding raw coordinates to a FFNN and using the output as wave function (simple NNWF).

V.1.2. Asymmetry analysis

Although we have now verified that our NNWFs approximate the desired ground state energies, we don’t know yet whether the approximation 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 $\langle \vec{x} \rangle$, $\langle \vec{x}^2 \rangle$ (Fig. 4) and the exchange symmetry measure (Fig. 5)

$$E_{\text{SM}}(\Psi) = \langle \Psi(x_1, x_2)|\Psi(x_2, x_1) \rangle / \langle \Psi(x)|\Psi(x) \rangle,$$  \hspace{1cm} (25)

which yields 1 for a symmetric $\Psi$, −1 for an antisymmetric $\Psi$ and 0 for a completely asymmetric $\Psi$. To complete the picture, we also computed the particle density $n(x) = \langle \sum_i \delta(x_i - x) \rangle$ (Fig. 6).

Looking at the results, it becomes immediately obvious that for certain Hamiltonian parameters $\alpha \gg 0$, $V \gg 0$ the observed exchange-symmetry and position expectation values do not match the expectations. 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 the same and zero, regardless of Hamiltonian parameters. In contrast though, for said choices of $\alpha$, $V$ we find the non-symmetric NNWFs to predict the two particles being located on opposite sides of the trap, i.e. $\langle x_1 \rangle = -\langle x_2 \rangle \neq 0$, and the symmetry measure is vanishing simultaneously. At the same time the energy $E_F$ and other expectation values like $\langle \vec{x}^2 \rangle$ and $n(x)$ remain close to expectation. These properties indicate that the NNWF 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 Fig. 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 $V \to \infty$, where the states can be exactly constructed from single-particle orbitals located on the left and right sides of the trap.

Concluding, we learned that if we chose to employ NNWFs without strict exchange-symmetry to describe indistinguishable particles, the VMC optimization of said 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 Eq. 25 to verify the desired symmetry, if it is of practical relevance. However, if exchange-symmetry of the NNWF can be guaranteed, e.g. by providing the used FFNN with exchange-symmetric input, the phenomenon discussed in this subsection is directly avoided.
FIG. 5. Exchange Symmetry Measure $ESM$ of all non-symmetric NNWFs, for the soft-core harmonic trap system.

FIG. 6. Density profiles $n(x)$ for the soft-core harmonic trap system, obtained with simple bosonic and fermionic product NNWFs. All values are denoted in atomic units.

V.2. H2 Molecule

For the hydrogen molecule, we considered both simple NNWF (in two different sizes) and simple bosonic product NNWF (as described in Section III.2), and obtained optimized energies for a range of protonic distances $R_{HH} = |R_1 - R_2|$. The resulting energy curve $E_T(R_{HH})$ represents an approximation to the Born-Oppenheimer potential of hydrogen and is displayed in Fig. 7(a), with remaining differences to reference energies $E_0(R_{HH})$ displayed in Fig. 7(b). Apparently, all employed NNWFs yield energy curves that qualitatively resemble the reference Born-Oppenheimer potential, with energy residuals ranging between $10^{-1}$ and $10^{-4}$ Ha, but differing by mostly 1-2 orders of magnitude between the two different NNWFs of same size. Despite the differences in energy accuracy, all NNWFs correctly predict the lowest energy.
to be at data point $R_{HH} = 1.4011 a_0$, with neighboring data points set at 1.36 and 1.44 $a_0$. This suggests that NNWFs might be especially useful in molecular geometry optimization. However, in contrast to the bosonic product NNWF, the simple NNWF of same size also predicts an energetically maximum at around $R_{HH} = 4.8 a_0$, an unphysical feature which is shared with other trial wave function formulations (ref?). However, as we can see from the results of the larger simple NNWF, that feature can be effectively suppressed by increasing the size of the FFNN.

In contrast to the results for the soft-core harmonic trap system, here we observe simple and simple bosonic product NNWF formulations achieving consistently and remarkably different energetic accuracy, when compared for the same FFNN size. We assume that this observation is not only a manifestation of singularities in the present Hamiltonian (TODO: Cusp), but also of the fact that the right side of our bosonic product NNWF formulations is varying with Hamiltonian parameter $R_{HH}$ in the case of $H_2^+$, while it is static in the case of the soft-core harmonic trap.

To complement the energy results, we display the ESM (Eq. 25) diagnostic for all employed NNWFs in Fig. 8. While all NNWFs learned approximately exchange-symmetric states for $R_{HH} < 3 a_0$, all NNWFs began to learn 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 in the case of 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), i.e. the NNWFs adopted a state of distinguishable particles.

VI. Conclusion

We have presented a VMC method using FFNN-based trial wave functions (NNWF) in continuous space to approximate the ground states of N-particle 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 wave function formulations 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 reasonable accuracy, 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 we considered the Hydrogen molecule in Born-Oppenheimer approximation, but still even the simple NNWF would yield a qualitatively adequate prediction of the Born-Oppenheimer potential.

We observed that the 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, however the practical relevance of this behavior remains questionable. We want to look at it again within future studies of many-electron systems and especially investigate methods for providing exchange-symmetric input to the FFNN, preventing the asymmetry inherently.

In conclusion, we have demonstrated that a NNWF represents a flexible trial wave function that can be straightforwardly applied to a variety of Hamiltonian settings, without necessarily requiring problem-specific adjustments. While these trial wave functions certainly do not compete in accuracy and efficiency with other modern methods when applied to few-electron systems, we suppose there might be a reasonable use of NNWFs for larger systems, complex Hamiltonians or whenever a very flexible trial wave function is required.

Despite the satisfying success of our method, there are clearly small improvements possible and probably necessary for application to realistic systems. On the technical side, we expect to achieve a significantly speed-up optimization process by employing more suitable differentiation techniques to compute the gradient, e.g. by a reverse-mode differentiation scheme. However, our currently used forward-accumulation scheme will most likely remain an efficient choice for the second order input derivatives, which are required to compute the exact kinetic energy of the NNWF. It should be mentioned that there are countless different ANN types, NNWF formulations or combinations with other methods thinkable, and we considered only a small subset of what is possible. Our energy-gradient-based approach to optimizing the NNWF is not the only choice thinkable. Overall, this leaves a lot of interesting opportunities for future research.

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, because the straightforward and general way of constructing (anti)symmetric NNWFs (Eq. 8) 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 that the optimization will make the NNWF approximately symmetric. This happens in other forms in all other QMC imaginary-time projection methods, like Diffusion Monte Carlo, and can be systematically improved by a more accurate tuning of the parameters (ref?).

When one is interested in the simulation of fermions however, the problem becomes more serious. In fact, the approach we used in this work introduces a fixed-node approximation. 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. In other words, we are hitting the fermion sign problem from a new perspective, and the hope to find a breakthrough that allows for unbiased and computationally affordable simulations is to be considered almost
null. However, due to the intrinsic flexible nature of the NN, there is hope to find a novel systematic approach to reduce the fixed-node bias.

Acknowledgments

A. Settling for a network structure

1. Activation Functions

Before we started production of data for this paper, we first investigated the influence of activation function choices on the achieved energy accuracy. Besides the $a_T$ and $a_G$ hidden layer activation functions mentioned in section II.2, we also tried to use the identity function for some of the hidden units:

$$a_I(p) = p$$

Besides the exponential activation function, for the output unit we also tried the commonly employed logistic activation function

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

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 the common choice of linear output activation spanning full $\mathbb{R}$, because in our testing we found it yielding unfavorable oscillations around value 0 for input ranges where the wave function has a small amplitude. Furthermore, in the bosonic case we would want to avoid a wave function with variable sign anyway.

Energy results for the harmonic soft-core trap and certain combinations of hidden layer and output activation functions are displayed in Fig. 9. Results for the combination of $a_T$ and $a_G$ hidden layer activation function with either $a_L$ or $a_E$ on the output are displayed in Fig. 10.

2. Hidden Layer Structure

Similarly, we investigated how the accuracy changes when changing the size and number of hidden layers. An overview of our results is shown in Fig. 11.

[1] W. L. McMillan, Phys. Rev. 138, A442 (1965).
[2] K. Hornik, Neural Networks 4, 251 (1991).
[3] Y. LeCun, Y. Bengio, and G. Hinton, Nature 521, 436 (2015).
[4] I. E. Lagaris, A. Likas, and D. I. Fotiadis, Computer Physics Communications 104, 1 (1997).
[5] G. Carleo and M. Troyer, Science 355, 602 (2017).
[6] X. Gao and L.-M. Duan, Nature Communications 8, 662 (2017).
[7] H. Saito and M. Kato, J. Phys. Soc. Jpn. 87, 014001 (2017).
[8] I. Glasser, N. Pancotti, M. August, I. D. Rodriguez, and J. I. Cirac, Phys. Rev. X 8, 011006 (2018).
[9] Z. Cai and J. Liu, Phys. Rev. B 97, 035116 (2018).
[10] Y. Levine, O. Sharir, N. Cohen, and A. Shashua, arXiv:1803.09780 [quant-ph] (2018), arXiv: 1803.09780.
[11] G. Carleo, Y. Nomura, and M. Imada, arXiv:1802.09558 [cond-mat, physics:physics, physics:quant-ph] (2018), arXiv: 1802.09558.
[12] M. Ruggeri, S. Moroni, and M. Holzmann, Phys. Rev. Lett. 120, 205302 (2018).
[13] W. K. Hastings, Biometrika 57, 97 (1970).
[14] Y. Nagaoka and T. Usui, Prog. Theor. Phys. E68, 392 (1968).
[15] D. P. Kingma and J. Ba, arXiv:1412.6980 [cs] (2014), arXiv: 1412.6980.
[16] P. Kocik and T. Sowiski, Scientific Reports 8, 48 (2018).
[17] K. Pachucki, Physical Review A 82 (2010), 10.1103/PhysRevA.82.032509.
FIG. 9. Energy differences to 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: T = hyperbolic tangent sigmoid, G = Gaussian, I = identity, E = exponential, L = logistic.

FIG. 10. Like Fig. 9, but comparing only T/G-L (LGS) and T/G-E (EXP) on a larger set of Hamiltonian parameters.
FIG. 11. Like Fig. 9, but comparing different numbers of hidden layers and hidden units for the case of T/G-E activation function configuration.